INVESTIGATION INTO THE ACCURACY AND PRACTICALITY OF METHODS FOR TRANSFORMING COORDINATES BETWEEN GEODETIC DATUMS

by

Andrew Carey Ruffhead, MPhil

A thesis submitted in partial fulfilment of the requirements of the University of East London for the degree of Doctor of Philosophy

School of Architecture, Computing and Engineering, University of East London

March 2021

ABSTRACT

This thesis is a study of methods of transforming coordinates between geodetic datums, the methods being generally known as datum transformations.

Direct methods are described and categorised as conformal, near-conformal and non-conformal. New variations on all three types are included in the direct methods: SMITSWAM (which avoids changes of coordinate-type), partially-conformal variants of Standard & Abridged Molodensky, and normalised generalisations of multiple regression equations (5 types). Reverse transformations are extensively covered, as are methods of derivation. In both cases, new algorithms are included.

Direct methods, with the exception of multiple regression equations, do not capture distortions in datum transformations. The thesis therefore includes a review of composite methods which extract a trend model and apply a surface-fitting technique (SFT) to the residuals. Sometimes the SFT is used as a gridding method, producing regularly-spaced data that can be interpolated as a final stage of the composite process.

The SFTs selected for detailed study include new variations on inverse-distance-to-a-power weighting and nearest-neighbour interpolation. These are called HIPFEAD and LIVONN respectively. In both cases, the variations are shown to have advantages in terms of accuracy of fit. Least-squares collocation and radial basis functions are shown to produce reusable vectors – described here as “revamped signals” – that enable interpolation without gridding.

Where the composite methods are used for gridding, it is shown that geodetic coordinates can be used, avoiding the need for projected grid coordinates. The interpolation options applied are piecewise-bilinear and piecewise-bicubic, the latter being an algorithm (believed to be new) that uses up to 12 “grid” points.

Case studies were considered using 6 datasets, two for Great Britain, one each for Western Australia, Ghana, Sweden and Slovenia. These showed beneficial properties of the new methods, both in the direct and composite categories. They also enabled comparisons of transformation methods generally.

ACKNOWLEDGEMENTS

The author wishes to express his thanks to the following:

- My Director of Studies Dr Brian Whiting, to whom I am indebted for his constant help, advice and encouragement throughout the investigation.
- My initial Second Supervisor Dr Darion Grant for his support and constructive suggestions.
- My subsequent Second Supervisor Mr Bryan Pearce for his support and constructive suggestions.
- Mr Richard Latham for showing me how to use the LSS contouring software package.
- Curtin University and Professor Joseph Awange in particular for the Western Australia dataset in Appendix C.
- Ordnance Survey and Mr Mark Greaves in particular for the Great Britain datasets in Appendix C.
- The Surveying & Mapping Authority of the Republic of Slovenia and Sandi Berk in particular for the Slovenia dataset in Appendix C.
- Mr Henry Rogers for persuading me that it was feasible to embark on a PhD in my late-60s.

TABLE OF CONTENTS

DEFINITIONS

anisotropy	state of having a physical property which has a different value when measured in different directions.
ArcGIS	Name of a "geographic information system for working with maps and geographic information" maintained by the Environmental Systems Research Institute.
basis functions	functions which are used in a linear combination to make a mathematical model.
geoid	hypothetical solid figure whose surface corresponds to mean sea level and its imagined extension under (or over) land areas.
geoid height	height of geoid above ellipsoid, so is dependent on geodetic datum.
geoid model	mathematical representation of the geoid for a specific area, or for the whole earth. Since it consists of geoid heights, it is dependent on geodetic datum.
gridding	process of converting data to data defined on a regular mesh.
gridded data	data defined at points on a regular mesh.
horizontal	in a plane tangential to a reference ellipsoid at a local point. This is a deliberately loose definition for use in this thesis and would not be appropriate in the context of physical geodesy.
isotropy	state of having a physical property which has the same value when measured in different directions.
L_1 norm	sum of magnitudes.
largest	quantity with the highest modulus. This use of the word "large" treats -996 as being larger than +995. It also treats 5, -5 and the complex number $3+4i$ as equally large.
level	see horizontal.
LSS	name of a "complete land survey, terrain modelling, volume, design and visualisation package" produced by McCarthy Taylor Systems Limited.
magnitude	size irrespective of sign.
monomial	product of non-negative powers, for example 1, x, x^2, xy, x^3y^4. This definition is adopted in preference to "polynomials with a single term" (which would include terms like $6x^3y^4$).

normalised	constrained to be in the range -1 to 1. For example, 0.5(x-14) is a normalised version of a variable x whose range of values is 12 to 16.
orthometric height	height above geoid.
partially-linear	description of a near-conformal 3D transformation in which the rotation matrix has been linearised but the linearised matrix as a whole is multiplied by the scale factor: $(1 + \Delta S)\begin{bmatrix} 1 & -R_Z & R_Y \\ R_Z & 1 & -R_X \\ -R_Y & R_X & 1 \end{bmatrix}$
polynomial	linear combination of monomials. Examples include $3 - x + 4.84576x^2 + 6.87519x^3$ and $x + 3.5y - xy + 6x^3y^4 + 7.5x^2y^5$. This definition allows powers of more than one variable.
rearrangement-type formula	reverse formula based on rearrangement. For example, the rearrangement-type formula for the inverse of $y = 2x^3 - 1$ is $x = [(y+1)/2]^{1/3}$.
rigid	type of transformation that preserves shape and size. In this thesis it allows for translation and rotation, but not reflection (because the context is geodetic datum transformations).
rotation	rotation in the sense of a position vector moving counter-clockwise when viewed from the positive side of the origin. This means that a clockwise rotation is negative in sign. (Some sources adopt the opposite convention.) In this thesis, rotation means transformation (overall change) rather than rotating motion.
same-formula inverse	inverse of a formula that uses the original form with different parameters.
shape parameter	parameter that controls the rate-of-change of a real-valued function: changing its value stretches or compresses the curve in the graphical representation.
simple same-formula inverse	same-formula inverse (usually approximate) which applies the parameters of the original formula with opposite signs.
smooth	having a continuous derivative (in the case of a function of one variable) or having continuous partial derivatives (in the case of a function of more than one variable).

vertical	parallel to the normal to a reference ellipsoid at a local point. This is a deliberately loose definition for use in this thesis and would not be appropriate in the context of physical geodesy. However, vertical and horizontal should be regarded as mutually perpendicular.
weighted average	linear combination of terms where the coefficients are non-negative and add up to 1.

ABBREVIATIONS

AGD84	Australian Geodetic Datum 1984.
AP/SE	absolute value of a parameter divided by its standard error.
ArcGIS	(See Definitions.)
BC	bicubic.
BL	bilinear.
BLM	bilinear model.
BT48	Borneo Triangulation 1948.
CF	coordinate frame.
C^n	continuous with continuous derivatives up to and including the *n*th order. Where the domain of the function has two or more dimensions, "derivatives" means partial derivatives.
CORS	Continuously Operating Reference Station.
DMA	US Defense Mapping Agency.
D48	Datum 1948 (used in Slovenia).
D96	Datum 1996 (used in Slovenia).
EEH	Equivalent-enlargement hypothesis.
ERLTO	eliminating ratios less than one.
ETRFnn	European Terrestrial Reference Frame 19nn. The full year is included for 2000 onwards.
ETRS89	European Terrestrial Reference System 1989.
FLBW	fully-linear Bursa-Wolf. For this version of the Bursa-Wolf transformation, the scaling is only applied to the diagonal of the linearised rotation matrix.
FLMB	fully-linear Molodensky-Badekas. For this version of the Molodensky-Badekas transformation, the scaling is only applied to the diagonal of the linearised rotation matrix.
GDA94	Geocentric Datum of Australia 1994.
GDBD2009	Geocentric Datum Brunei Darussalam 2009.
GGD	Georgia Geodetic Datum.
GNSS	Global Navigation Satellite Systems.
GPS	NAVSTAR Global Positioning System.
GRS67	Geodetic Reference System 1967.
GRS80	Geodetic Reference System 1980 or Reference Ellipsoid 1980.
HICFEAD	hybrid inverse cubic function embodying accelerated decline.

HIPFEAD	hybrid inverse power function embodying accelerated decline.
HISFEAD	hybrid inverse square function embodying accelerated decline.
IAMVAM	inverse Abridged Molodensky via applied misclosure.
IAMVVAM	inverse Abridged Molodensky variation via applied misclosure.
ICSM	Intergovernmental Committee on Surveying and Mapping.
IERS	International Earth Rotation and Reference Systems Service.
IMQ	inverse multiquadric.
ISMVAM	inverse Standard Molodensky via applied misclosure.
ISMVVAM	inverse Standard Molodensky variation via applied misclosure.
ITRFnn	International Terrestrial Reference Frame 19nn. The full year is included for 2000 onwards.
ITRS	International Terrestrial Reference System.
IUGG	International Union of Geodesy and Geophysics.
LIVONN	linear interpolation variant on nearest neighbour.
LSC	least-squares collocation.
LSS	(See Definitions.)
MDNN	median distance to nearest neighbour, taken over a set of data points.
MDZ	median distance to zero, taken over a set of data points.
ML	multilog.
MQ	multiquadric.
MRE	multiple regression equation.
NAD27	North American Datum 1927.
NGA	US National Geospatial-Intelligence Agency.
NIMA	US National Imagery and Mapping Agency.
NCS	natural cubic spline.
NSB	not strictly bounded. An NSB interpolating function can take values outside the range given at data points.
OSGB36	Ordnance Survey Great Britain 1936.
ppm	parts per million.
PCV	Partially-conformal variation
PV	position vector.
PLBW	partially-linear Bursa-Wolf. For this version of the Bursa-Wolf transformation, the scaling is applied to the whole of the linearised rotation matrix.

PLMB	partially-linear Molodensky-Badekas. For this version of the Molodensky-Badekas transformation, the scaling is applied to the whole of the linearised rotation matrix.
RBF	radial basis function.
RD	residual distance. This will be same as positioning error if the residuals are errors.
RHD	residual horizontal distance. This will be same as horizontal positioning error if the residuals are errors.
r_{max}	limit-of-influence. Distance beyond which data has no influence.
RMS	root-mean-square.
RT90	Rikets koordinatsystem 1990.
SAD69	South American Datum 1969.
SFI	same-formula inverse. The original formula is used with different parameters.
SFT	surface-fitting technique.
SI	simple inverse (used when SSFI isn't quite appropriate).
SMITSWAM	Standard Molodensky In Two Stages With Applied Misclosure.
SSFI	simple same-formula inverse. The original formula is used but the signs of the original parameters are reversed.
SWEREFyy	Swedish Reference Frame (where yy is 93 or 99).
TPC	thin plate spline.
TRF	Terrestrial Reference Frame.
TRS	Terrestrial Reference System.
VBA	Visual Basic for Applications.
WGS72	World Geodetic System 1972. This is the name of the ellipsoid as well as the name of the datum.
WGS84	World Geodetic System 1984. This is the name of the ellipsoid as well as the name of the datum.
3PC	3-parameter conformal.
4Q	4-quadrant.
7PC	7-parameter conformal.

Part One: Introduction

CHAPTER 1: GENERAL CONCEPTS AND OVERVIEW

1.1 Background

Our knowledge of the earth's shape and size is fundamental to mapping and the description of position worldwide. This knowledge has greatly improved over time resulting in numerous models of the earth and mapping coordinate systems. Today, with the advent of satellite positioning systems, the determination of a global model of the earth is possible. It is also not difficult to establish the position of a point using this technology. However, relating it to measurements made in the past is, to say the least, problematic. The mapping in many countries has been associated to different earth models over time, and the coordinates of a point will differ according to the model adopted at the time.

This study is about mathematical relationships between positioning systems. They arise from a multitude of positioning systems or *datums* that are in use for precise mapping and navigation. A *datum transformation* is the process of converting the coordinates of points from one earth model to another. It requires knowledge of the individual models and the coordinates of points established in that model. There are several ways in which the transformation can take place and these are investigated here.

An example can be found in the mapping of Great Britain which was started in 1747. The first network of points, the principle triangulation, came about during 1791-1853. The original network was a triangulation with all angles measured but only one distance (5.190 miles, equivalent to 8.352km). The network was ill-conditioned, so a new re-triangulation took place during 1936-1962; the computation was done in parts, due to the lack of computing power. Since then, electronic measurement of distances has become possible, satellite positioning techniques have been developed, and the ability to compute large systems of equations has become easy. These have identified great weaknesses in the GB mapping system that users of the data were unaware of and have not affected their use. The choice was therefore to republish all the mapping or to provide a means to overcome these issues when necessary. Ordnance Survey, after consultation with its users, produced ways of transforming the older 1936-based coordinates to comply with satellite positioning.

In general, older datums around the world were defined in terms of local areas, typically an individual country or a group of countries. Positions were determined from astronomical observations and by traditional survey techniques (distance & angle measurement and

levelling) followed by computations designed to minimise the effect of measurement errors. Satellites have revolutionised surveying methods, leading to

- increasingly precise knowledge about the size & shape of the earth and its gravity field, and
- increasingly accurate measurement of positions.

This has led to the definition of new datums with global coverage.

The use of Global Navigation Satellite Systems (GNSS) is now commonplace, with almost every mobile telephone and tablet including a receiver capable of determining global position to a few metres in its datum. Without correction or transformation, the displayed position will not relate to current mapping when the latter is in another datum (which is often the case).

Precise GNSS receivers are in common use by the surveying profession as techniques have improved to give centimetre precision in real-time. As a result, there has been an increasing requirement for surveys to be connected to the national or project coordinate systems which are defined by the use of Continuously Operating Reference Stations (CORSs). For example, Ordnance Survey has established OS Net, a network of some 120 CORSs; this provides Ordnance Survey's own surveying teams with access to the new national datum, ETRS89, with data for postprocessing available free on its website, and with real-time data available via commercial partners to users in the field. However, the mapping data continues to be in another datum, OSGB36, and so transformation is needed to combine their use.

In addition to the requirement noted above, datum transformations are needed for the following purposes:

- Positions known in a local datum sometimes need to be transformed into positions in a global datum for use in navigation systems.
- Positions determined by satellites in a global system sometimes need to be transformed into positions in a local datum for use by a national mapping agency.
- Positions known in one local datum need to be computed in terms of another local datum if a country wishes to adopt the latter as a replacement datum.

Modelling datum transformations is an inexact science, for reasons discussed in Section 1.6.

Research into datum transformations has given rise to thousands of papers from geoscientists around the world. An overview of the current status of research is given in Section 1.8.

1.2 Aims and objectives

Foremost among the aims was a comprehensive overview of geodetic datum transformations that introduced new methods and compared them with existing methods. The purpose of the former was to improve on existing methods for accuracy and/or ease of computation.

The scope included methods of inverting transformations (applying transformations in the reverse direction) with the aim of introducing new ones and improving existing ones. The scope also included methods of deriving transformations (from sets of points common to two datums) with the aim of introducing new ones where existing methods were problematic.

In practice, the "optimum" transformation is often determined by the accuracy required for a particular purpose and the most practical method to achieve it. The thesis compares different transformations from six case studies for accuracy and practicality.

Accuracy is measured by root-mean-square (RMS) of residuals. Where there are sufficient data points for some to be set aside for independent verification, the RMS is computed from test points. Other measures of probable accuracy include the extent or absence of extrapolation effects and the model behaviour as illustrated in contour maps. Practicality is mainly about ease of computation. This affects derivation of transformation as well as application. Far more transformations with 3 or 7 parameters have been derived than those The growing number of derived transformations based on advanced surface-fitting techniques is still far fewer than simpler models with 3 or 7 parameters (see, for example, ESRI [2012]) so the latter are stronger candidates for software tools with worldwide application.

Besides comparing existing transformation methods, the case studies aim is to confirm that the new transformation methods have advantages over the methods they are adapted from: either greater accuracy without a major increase in computation or a more economical route to the same accuracy. In the case of Great Britain (2D), for example, accuracy levels were sought that improved on the 0.1m level obtainable from OSTN15.

The case studies are based on 6 datasets, two for Great Britain, one each from Western Australia, Ghana, Sweden and Slovenia. Although availability of data was a consideration, the selection was designed to reflect the variety of transformation requirements that occur in

practice. Hence the differences in size-of-region, coverage-of-region, number of points and presence (or otherwise) of heights.

The case studies include the following:

- accuracy statistics of the transformations derived;
- comparisons with published transformations (where available) for supplementary verification;
- observations on aspects of the results which are related to distinctive aspects of the data (*eg* mathematically-generated heights in Ghana);
- modifications of transformations where made necessary by characteristics of the data (*eg* clusters of control points in Australia and Great Britain);

Section 1.8 reviews the literature that describes existing knowledge about datum transformations, although further evaluation of relevant publications is provided within particular sections to aid understanding. Section 1.9 provides details of how this study builds on existing knowledge.

1.3 The geodetic datum

Before proceeding further, the basic concepts need a formal introduction.

A geodetic datum is a system for precisely measuring locations on Earth (or other planetary body). An alternative name which has increased in usage is terrestrial reference system (TRS).

Geometrically speaking, a geodetic datum is a representation of the earth that includes an ellipsoid of revolution about a polar (Z) axis and axes (X and Y) in the equatorial plane such that the positive X-axis defines zero longitude. The ellipsoid is fitted to the earth according to whether the datum is intended for local or global use. For a local datum, the ellipsoid is designed to fit the geoid closely in the local area. For a global datum, the ellipsoid is designed to give the best possible fit to the global geoid. Either way, the definition of the datum must define (either directly or indirectly)

- the size, shape and centring of the ellipsoid,
- the direction of the polar axis, and
- the zero meridian.

Figure 1-1 shows the axes used for Cartesian coordinates (X, Y, Z). The geodetic coordinates (ϕ, λ, h) are shown in Figures 1-1 and 1-2. It should be noted that h is individually known as ellipsoidal height rather than geodetic height.

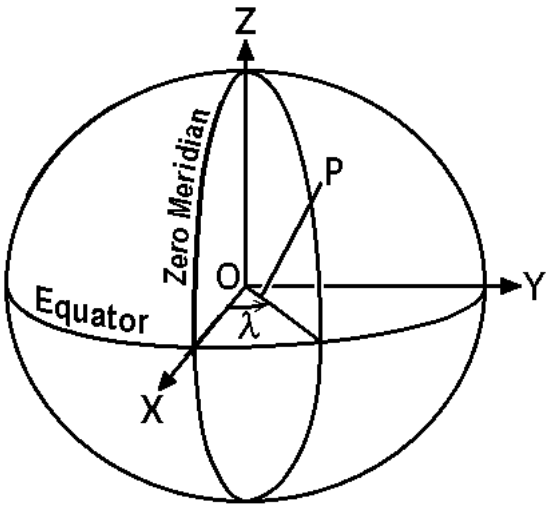


Figure 1-1: Ellipsoid showing Cartesian axes, the ellipsoidal normal through point P and the geodetic longitude (λ) of P.

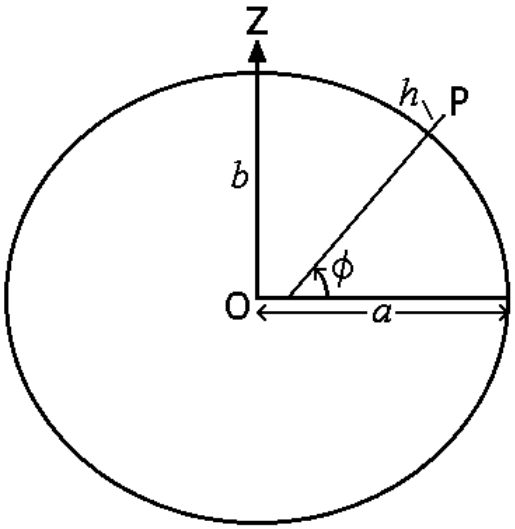


Figure 1-2: Meridian plane containing point P showing the ellipsoidal normal through P, the geodetic latitude (ϕ) of P and the ellipsoidal height (h) of P.

Table 1-1: Ellipsoidal notation conventions

a	Semi-major axis
b	Semi-minor axis
f	Flattening: (a-b)/a
e	Eccentricity: $\sqrt{a^2 - b^2}/a$ so that $e^2 = (a^2 - b^2)/a^2$ and $e^2 = 2f - f^2$.
e'	Second eccentricity: $\sqrt{a^2 - b^2}/b$
ε	$(e')^2$ so that $\varepsilon = (a^2 - b^2)/b^2 = e^2/(1 - e^2)$.

Global datums are geocentric, which is to say that the centre of the reference ellipsoid is the earth's centre of gravity. Only in the era of satellite surveying has it been possible to identify where that point is. In recent years, some geocentric ellipsoids have been adopted for local datums to replace older ones.

A concept closely related to geodetic datums is the "terrestrial reference frame" (TRF) consisting of a network of reference points with known coordinates. They are sometimes referred to as datums, but it is more accurate to regard them as datum realisations. They are tied to the point in time (or epoch) when positions in the network were computed, on the basis that slight changes can occur due to tectonic-plate movement. (This is not a consideration when a datum is defined in terms of a region which is tied to a tectonic plate.) The relationships between datum

realisations – and between a geodetic datum and a geodetic datum realisation – can also be considered as datum transformations.

There are hundreds of geodetic datums and datum realisations. Table 1-2 contains those referred to in this thesis. Similarly, Table 1-3 lists the ellipsoids referred to in the thesis.

Table 1-2: Selected Datums And Their Ellipsoids

Datum (or Realisation)	Ellipsoid
Adindan	Clarke 1880 Modified
Accra	War Office 1924 (aka War Office 1926)
Australian Geodetic Datum 1984 (AGD 84)	Australia National (1966)
Arc 1950	Clarke 1880 (Arc)
Arc 1950 (Zaire)	Clarke 1880 Modified
BT48	Everest (Borneo)
Córrego Alegre	International 1924
Croatian State Coordinate System (HDKS)	Bessel 1841
Croatian Terrestrial Reference System (HTRS96)	Reference Ellipsoid 1980 (GRS80)
D48	Bessel 1841
D96 [name adopted by Slovenia for ETRS89]	Reference Ellipsoid 1980 (GRS80)
European Terrestrial Reference System 1989 (ETRS89)	Reference Ellipsoid 1980 (GRS80)
Fatu Iva 1972	International
Genova 1902	Bessel 1841
Geocentric Datum of Australia 1994 (GDA94)	Reference Ellipsoid 1980 (GRS80)
Geocentric Datum Brunei Darussalam 2009 (GDBD2009)	Reference Ellipsoid 1980 (GRS80)
Georgia Geodetic Datum (GGD)	Reference Ellipsoid 1980 (GRS80)
Indian	Everest (India)
International Terrestrial Reference Frame 1996 (ITRF96)	Reference Ellipsoid 1980 (GRS80)
Ireland 1965 (aka Geodetic Datum of 1965)	Airy Modified
Leigon	Clarke 1880 Modified
North American Datum 1927 (NAD27)	Clarke 1866
North American Datum 1983 (NAD83)	Reference Ellipsoid 1980 (GRS80)
Ordnance Survey Great Britain 1936 (OSGB36)	Airy (1830)
Pulkovo 1942	Krassovsky (1940)
Réseau Géodésique de la Réunion 1992 (RGR 1992)	Reference Ellipsoid 1980 (GRS80)
Réunion 1947	International 1984
Rikets koordinatsystem 1990 (RT90)	Bessel 1841
South American Datum 1969 (SAD69)	GRS67 (Truncated)
SWEREF93	Reference Ellipsoid 1980 (GRS80)
World Geodetic System 1972 (WGS72)	World Geodetic System 1972 (WGS72)
World Geodetic System 1984 (WGS84)	World Geodetic System 1984 (WGS84)

Table 1-3: Ellipsoids And Their Parameters

Ellipsoid	a (in metres)	e^2	$1/f$	b (in metres)
Airy (1830)	6377563.396	0.00667054000000	299.3249647	6356256.90924
Airy Modified	6377340.189	0.00667054000000	299.3249647	6356034.44794
Australia National (1966)	6378160.000	0.00669454185000	298.2500000	6356774.71921
Bessel 1841	6377397.155	0.00667437223000	299.1528128	6356078.96282
Clarke 1866	6378206.400	0.00676865800000	294.9786982	6356583.79999
Clarke 1880 (Arc)	6378249.145	0.00680348102000	293.4663077	6356514.96639
Clarke 1880 Modified	6378249.145	0.00680351128000	293.4650000	6356514.86956
Everest (Borneo)	6377298.556	0.00663784663000	300.8017000	6356097.55030
Everest (India)	6377301.243	0.00663784607000	300.8017255	6356100.23016
GRS67 (Truncated)	6378160.000	0.00669454185000	298.2500000	6356774.71921
International	6378388.000	0.00672267002000	297.0000000	6356911.94614
Krassovsky (1940)	6378245.000	0.00669342162000	298.3000000	6356863.01878
Reference Ellipsoid 1980 (GRS80)	6378137.000	0.00669438002290	298.2572221	6356752.31414

South American 1969	6378160.000	0.00669454185000	298.2500000	6356774.71921
War Office 1924 (aka War Office 1926)	6378300.000	0.00674534332000	296.0000000	6356751.68918
World Geodetic System 1972 (WGS72)	6378135.000	0.00669431778000	298.2600000	6356750.52001
World Geodetic System 1984 (WGS84)	6378137.000	0.00669437999013	298.2572236	6356752.31425

Datums that predate the satellite era had limitations for 3-dimensional positioning. Traditional surveying methods for determining latitude and longitude were very different to the levelling methods used to determine orthometric heights. The latter are heights above the geoid which is a physical entity, albeit a theoretical one that approximates mean sea level and extends it into land areas. Orthometric heights can only be converted to ellipsoidal heights if there is a geoid model providing the separation between ellipsoid and geoid (called geoid height). Ellipsoidal heights obtained in this way are far less accurate than those obtained by GNSS for geocentric reference ellipsoids.

The Geodetic Reference System 1980 (GRS80) has a special significance. It was adopted by the International Union of Geodesy and Geophysics (IUGG) at its XIV General Assembly in December 1979. It was designed to represent the size, shape and gravity field to an accuracy adequate for geodetic, geophysical, astronomical and hydrographic applications. The GRS80 ellipsoid is a part of that system.

The World Geodetic System 1984 (WGS84) was defined by the US National Imagery and Mapping Agency (NIMA) as a geocentric datum. The reference ellipsoid, also called WGS84, is a variant of GRS80 (same value of a, a difference of 0.00011m in b) and its orientation was aligned to the orientation given in 1984 by the forerunner of the International Earth Rotation and Reference Systems Service (IERS). The IERS alignment has been updated in periodic re-definitions of WGS84 as a datum realisation (or TRF). The realisations of WGS84 are described in detail in NGA (2014).

IERS produces the International Terrestrial Reference System (ITRS) which is – in a loose sense – a geodetic datum or TRS whose reference ellipsoid is GRS80. In practice, points are positioned in terms of realisations or TRFs. Each one is named International Terrestrial Reference Frame nnnn, which is shortened to ITRFnn for those with Epoch 19nn.0 and ITRF20nn for those with Epoch 20nn.0.

Some of the newer datums intended for mapping a region are based on a particular ITRF, so that the effect of tectonic motion is removed. Examples include North American Datum 1983, Geocentric Datum of Australia 1994 and European Terrestrial Reference System 1989

(ETRS89). The latter is called a TRS rather than a TRF, and this reflects the fact that has its own realisations (*eg* ETRF89, ETRF2000, ETRF2014).

It is important to draw a distinction between datum transformation methods and coordinate conversion methods. In this paper, a datum transformation converts coordinates in one datum to coordinates in another (*eg* $\phi_{NAD27}, \lambda_{NAD27}, h_{NAD27} \rightarrow \phi_{WGS84}, \lambda_{WGS84}, h_{WGS84}$). The phrase "coordinate conversion" implies a conversion of one type of coordinate to another (*eg* ϕ, λ, $h \rightarrow X$, Y, Z) *within* a particular datum.

1.4 Coordinate types within a datum

The previous section introduced geodetic and Cartesian coordinates within a datum. This section describes the relationship between them and their relationship to other types of coordinates within a geodetic datum.

1.4.1 Geodetic and Cartesian coordinates

Geodetic and Cartesian coordinates are the most commonly used 3-dimensional coordinates.

The equations for converting geodetic coordinates to geocentric Cartesian coordinates are as follows.

$$\nu = \frac{a}{\sqrt{1-e^2 \sin^2 \phi}}. \tag{1-1}$$

$$X = (\nu + h) \cos \phi \cos \lambda. \tag{1-2}$$

$$Y = (\nu + h) \cos \phi \sin \lambda. \tag{1-3}$$

$$Z = [\nu(1 - e^2) + h] \sin \phi. \tag{1-4}$$

The derivation of these equations can be found, for example, in Section 5-3 of Heiskanen and Moritz (1967).

The reverse process for converting Cartesian coordinates to geodetic coordinates is less straightforward. The oldest-known process is the iterative approach that is recommended in Section 5-3 of Heiskanen and Moritz (1967). Ruffhead (2016) expressed the algorithm in terms of the distance (r) from the polar axis:

$$r = \sqrt{X^2 + Y^2}; \tag{1-5}$$

$$\lambda = \text{arctan2}[X, Y] \tag{1-6}$$

unless $r < a/10^{12}$, in which case λ should be set to 0. (Longitude is actually indeterminate when X and Y are both zero.) Figure 1-3 shows the range of the arctan2 function.

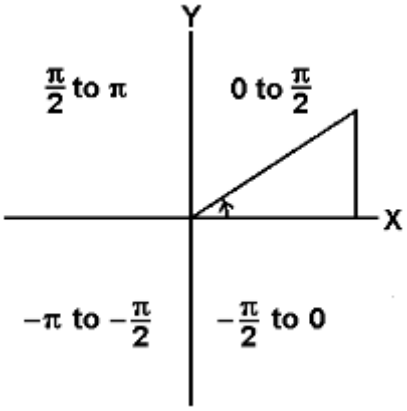


Figure 1-3: The four-quadrant arctan2 function.

Unless $|Z| < a/10^{12}$ (in which case ϕ should be set to 0) the following iterative approach is used to compute ϕ.

$$\phi = \arctan\left[\frac{Z}{(1-e^2)r}\right] \text{ as a first estimate.} \tag{1-7}$$

$$\phi' = \phi, \text{ so that } \phi' \text{ is a copy that can be compared with the next estimate.} \tag{1-8}$$

$$\nu = \frac{a}{\sqrt{1-e^2\sin^2\phi'}}. \tag{1-9}$$

$$\phi = \arctan\left[\frac{Z+e^2\nu\sin\phi'}{r}\right]. \tag{1-10}$$

Assignments (1-8) to (1-10) are performed up to 5 times until $|\phi-\phi'|<10^{-12}$radians.

If $|\phi|<\pi/4$,

$$h = \frac{r}{\cos\phi} - \nu, \tag{1-11}$$

otherwise

$$h = \frac{Z}{\sin\phi} - \nu \cdot (1 - e^2). \tag{1-12}$$

There are two reasons for the "5 times" limit on latitude computation. Firstly, it prevents the possibility that cumulative round-off error might prevent the measured difference in approximations from falling below the specified tolerance. Secondly, the sufficiency of the limit has been established by the convergence tests carried out for this study; these covered the full range of latitudes and 3 levels of heights (see Figure 1-4). The 5th iteration is only needed for full coverage of aerial points.

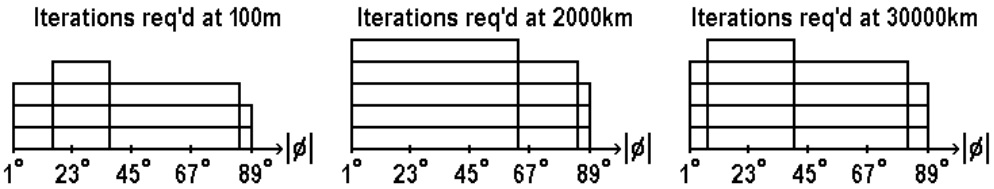


Figure 1-4: Iterations required for Cartesians-to-latitude conversion for the full range of latitudes at height levels 100m, 2000km and 30000km.

A near-exact alternative method of converting Cartesians to latitude was proposed by Bowring (1976). This uses the semi-minor axis b and the quantity $\varepsilon = e^2/(1\text{-}e^2)$. The calculation-of-latitude stage is replaced by

$$u = \arctan[(Z/r) \cdot (a/b)] \,, \tag{1-13}$$

where r is defined by (1-5), and

$$\phi = \arctan[(Z + \varepsilon b \sin^3 u)/(r - e^2 a \cos^3 u)]. \tag{1-14}$$

Fukushima (1999) recommended a variation in the implementation of Bowring's method. Based on trigonometric identities, it is designed to optimise the computation of latitude from Cartesians. The calculation-of-latitude stage is replaced by

$$T = (Z/r) \cdot (a/b), \tag{1-15}$$

where r is defined by (1-5),

$$C = \frac{1}{\sqrt{1+T^2}} \,, \tag{1-16}$$

$$S = CT, \tag{1-17}$$

and

$$\phi = \arctan[Z + \varepsilon b S^3)/(r - e^2 a C^3)]. \tag{1-18}$$

Since 1970, analytical methods that avoid iteration have been proposed by, for example, Paul (1973), Borkowski (1989), and Vermeille (2002, 2004, 2011). They are, of course, only exact theoretically because rounding-off always occurs in their implementation. An extensive list of studies on (ϕ, λ, h)↔(X, Y, Z) coordinate conversions can be found in Featherstone and Claessens (2008).

For the height computation, the test $|\phi|<\pi/4$ is one of many that can be used to avoid a small denominator. Between latitudes ±84°, $|\cos\phi|$ will exceed 0.1, so (1-11) could be used every time.

1.4.2 Grid coordinates

For some purposes, geodetic coordinates (ϕ, λ) are converted to plane coordinates (eastings and northings) by a map projection. Among the recognised projections are Bonne, Cassini-Soldner, Laborde, Lambert Conformal Conic, Mercator and Transverse Mercator (including Universal Transverse Mercator or UTM).

Some authors use (E, N) to denote grid coordinates in the eastern and northern directions, which avoids ambiguity provided E and N are not being used for other purposes. Other authors use (x, y) and don't always make clear whether x is measured in the eastern or northern direction. Snyder (1987) specifies the X-axis as pointing in the eastern direction and the Y-axis as pointing in the northern direction. He does note, however, that "Many British texts use X and Y axes interchanged, not rotated, from this convention".

For the most popular map projections, there are highly accurate inverse algorithms for converting grid coordinates back to (ϕ, λ). A wide range is covered by Snyder (1987).

For some lesser-known projections, the inverse algorithm is not very accurate. In such cases, the approximate geodetic coordinates can be improved by a generalisation of the Newton-Raphson method for solving non-linear equations. This involves the inversion of a matrix of partial derivatives. This approach was first proposed by Ruffhead (1998) who applied it to the Syrian Stereographic projection (Thomas, 1947). It was subsequently applied to three commonly-used cartographic projections by Ipbüker and Bildirici (2002), and then to several pseudo-cylindrical projections by Ipbüker (2009).

1.4.3 Local Cartesian coordinates

In the case of a local geodetic datum, it is sometimes convenient to convert the geocentric Cartesian coordinates (X, Y, Z) to Cartesian coordinates (X', Y', Z') with a local point P_0 as origin. As shown in Figure 1-5, the axes of the local Cartesian coordinates are parallel to the geocentric axes.

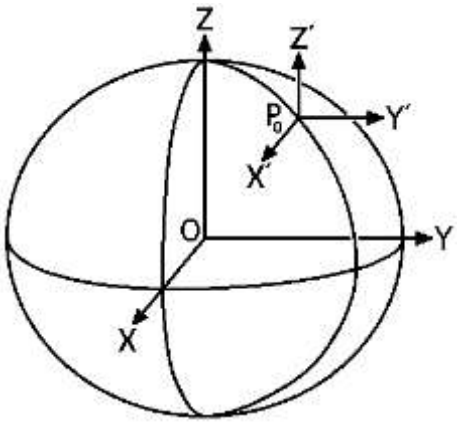


Figure 1-5: Axes of local Cartesian coordinates at P_0.

Given coordinates (X_0, Y_0, Z_0) at P_0, the relationship between geocentric and local Cartesian coordinates is given by

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix}. \tag{1-19}$$

1.4.4 Local level coordinates

In the case of a local geodetic datum, it is sometimes convenient to convert the geocentric Cartesian coordinates (X, Y, Z) to coordinates (X', Y', Z') in a "local level system". A local point P_0 is chosen as origin and the Z'-axis is the ellipsoidal normal to the ellipsoid through that point. The X'-axis points east along the parallel of latitude through P_0 and the Y'-axis points north along the median through P_0, as shown in Figure 1-6. Some sources differ in their description of the system; Cross *et al* (1982, page 8) uses the term "topocentric, vertical coordinates" and has the local X-axis pointing north.

Vertical lines in the locality of P_0 have approximately the same direction as the Z'-axis, so the perpendicular plane containing the X'-axis and Y'-axis is approximately horizontal. It may be considered a quasi-horizontal plane, hence the term "local level system".

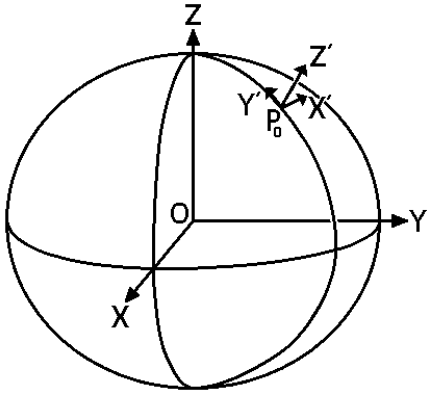


Figure 1-6: Axes of local level coordinates at P_0.

The coordinate conversion process can be regarded as three stages:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \rightarrow \begin{bmatrix} X_{(1)} \\ Y_{(1)} \\ Z_{(1)} \end{bmatrix} \rightarrow \begin{bmatrix} X_{(2)} \\ Y_{(2)} \\ Z_{(2)} \end{bmatrix} \rightarrow \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} \tag{1-20}$$

where the intermediate vectors contain the coordinates computed from the 1st and 2nd stages, hence the subscripts "(1)" and "(2)".

The first stage is a translation based on the Cartesian coordinates (X_0, Y_0, Z_0) of P_0:

$$\begin{bmatrix} X_{(1)} \\ Y_{(1)} \\ Z_{(1)} \end{bmatrix} = \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \tag{1-21}$$

The second stage is a counter-clockwise rotation of $\pi/2 + \lambda_0$ about the new *Z*-axis, making the *X*-axis point east and bringing the *Y*-axis into the north-south plane containing the ellipsoidal normal (so that the normal makes an angle ϕ_0 with the negative *Y*-axis):

$$\begin{bmatrix} X_{(2)} \\ Y_{(2)} \\ Z_{(2)} \end{bmatrix} = \begin{bmatrix} \cos(\pi/2 + \lambda_0) & \sin(\pi/2 + \lambda_0) & 0 \\ -\sin(\pi/2 + \lambda_0) & \cos(\pi/2 + \lambda_0) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{(1)} \\ Y_{(1)} \\ Z_{(1)} \end{bmatrix} \tag{1-22}$$

The third stage is a counter-clockwise rotation of $\pi/2 - \phi_0$ about the new *X*-axis (which may be easier to visualise as a clockwise rotation of $\pi/2 - \phi_0$ about the negative *X*-axis), bringing the *Z*-axis into the ellipsoidal normal and making the *Y*-axis point north:

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\pi/2 - \phi_0) & \sin(\pi/2 - \phi_0) \\ 0 & -\sin(\pi/2 - \phi_0) & \cos(\pi/2 - \phi_0) \end{bmatrix} \begin{bmatrix} X_{(2)} \\ Y_{(2)} \\ Z_{(2)} \end{bmatrix} \tag{1-23}$$

From these stages it is easily shown that

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} -\sin\lambda_0 & \cos\lambda_0 & 0 \\ -\sin\phi_0\cos\lambda_0 & -\sin\phi_0\sin\lambda_0 & \cos\phi_0 \\ \cos\phi_0\cos\lambda_0 & \cos\phi_0\sin\lambda_0 & \sin\phi_0 \end{bmatrix} \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \tag{1-24}$$

Hofmann-Wellenhof and Moritz (2006) notes that the coordinates are sometimes denoted as ENU coordinates, indicating east, north and up. It is therefore surprising that they swap the horizontal components, so that north precedes east. As a result, their conversion matrix for global-to-local conversion is

$$\begin{bmatrix} -\sin\phi_0\cos\lambda_0 & -\sin\phi_0\sin\lambda_0 & \cos\phi_0 \\ -\sin\lambda_0 & \cos\lambda_0 & 0 \\ \cos\phi_0\cos\lambda_0 & \cos\phi_0\sin\lambda_0 & \sin\phi_0 \end{bmatrix}.$$

(The transpose is given in their equation 5-66 as the conversion matrix for local to global coordinates.)

The reverse process to (1-24) can be computed exactly from the following equation:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} + \begin{bmatrix} -\sin\lambda_0 & -\sin\phi_0\cos\lambda_0 & \cos\phi_0\cos\lambda_0 \\ \cos\lambda_0 & -\sin\phi_0\sin\lambda_0 & \cos\phi_0\sin\lambda_0 \\ 0 & \cos\phi_0 & \sin\phi_0 \end{bmatrix} \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} \quad (1\text{-}25)$$

To preserve right-hand axes, this study has opted to have the X'-axis pointing east and the Y'-axis pointing north, in accordance with Figure 1-6. This is in line with the convention for the "local geodetic frame" adopted by Soler (1976, page 7). Not surprisingly, the rotation matrix converting "geocentric to local system" (ibid, page 14) matches that of (1-25).

Soler and Hothem (1988, page 90) adopts (e, n, u) as the local geodetic frame coordinates, ensuring a right-handed system. Their specification of the origin is "any point P(λ, ϕ, h) referred to a given ellipsoid" (ibid, page 90). This generic assumption is adopted here. As a result, P_0 might be a point on the reference ellipsoid (as in Figure 1-6) or a ground point (*ie* on Earth's surface). It might even be a point inside the reference ellipsoid, as in Andrei (2006), in which case the quasi-horizontal plane through P_0 intersects the ellipsoid.

Veis (1960, page 125) defines local coordinates in the same way as Soler and Hothem, the origin being a ground point projected onto the ellipsoid via the ellipsoidal normal. Furthermore, Veis's equation 86 (ibid, page 125) is equivalent to equation (1-24) above. However, some authors were determined to define a right-handed system in a different way with X' representing the north-south direction. Rapp (1993, page 68) and Leick (1995, page 480) define the Veis system to be south, east, up. (That is actually Veis's system rotated by 90° about the upwards axis.)

1.5 Coordinate conversions within the datum transformation process

Coordinate transformations between datums sometimes involve only a single type of position coordinates in each datum. In such cases the datum transformation is a direct process, as illustrated in Figure 1-7.

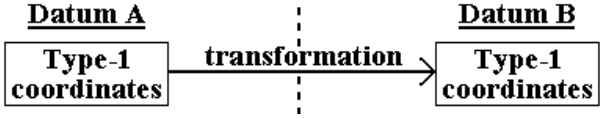


Figure 1-7: Direct transformation, involving only one type of position coordinates.

However, there are occasions when datum transformation models cannot be applied directly to the type of coordinates being transformed. In such cases a multi-stage process is needed,

as illustrated in Figure 1-8. Coordinate conversions need to be applied both before and after the transformation.

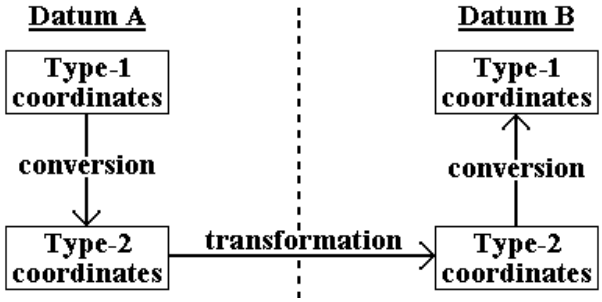


Figure 1-8: Multi-stage transformation process, involving more than one type of position coordinates.

A typical example is when the object is to transform geodetic coordinates from Datum A to Datum B and the transformation model is based on Cartesian coordinates. The geodetic coordinates in Datum A are converted to Cartesian coordinates before the transformation. The transformed Cartesian coordinates are converted to geodetic coordinates in Datum B.

1.6 Transformation requirements

The purpose of a datum transformation is to provide a mathematical or computational means to transform the coordinates of a point from one datum to another. In general, this is not an exact science, because the differences in coordinates at known points between datums are largely a reflection of different methods of measurement. This is due partly to the limitations of pre-satellite methods and partly due to different satellite methods (*eg* Doppler, GPS). What might be a suitable type of transformation between one pair of datums might be totally unsuitable between a different pair.

Another reason for different types of transformation is variation in accuracy requirements. For accuracy within a few metres, a simple model may suffice and may be preferred for ease of implementation. For the maximum attainable accuracy, something more complicated is usually needed.

The type of transformation needed is sometimes influenced by the size and shape of the area common to both datums. A model which is a function of (ϕ, λ), for example, may need more latitude-dependent terms than longitude-dependent terms if the north-south extent greatly exceeds the east-west extent.

Sometimes the data points common to both datums only cover a relatively small fraction of the area for which the transformation is required, possibly because of historic survey data being lost. In this instance, transformations with a limited extrapolation effect are the safest choice.

Some transformations only involve two-dimensional coordinates; others involve three-dimensional coordinates. The distinction is not entirely clear-cut as Cartesian coordinates (X, Y, Z) can be used to transform horizontal coordinates (ϕ, λ) if dummy height values are introduced to facilitate coordinate conversions.

One type of transformation is the *conformal* transformation which preserves shape: angles between lines are unchanged and distances change – if at all – by the same proportional amount.

A special case is the *rigid* transformation which preserves size and shape: angles between lines and distances are unchanged.

Some transformations are *near-conformal* and, if assumptions behind approximations are valid, they are virtually indistinguishable from conformal transformations. Examples include Standard Molodensky, Abridged Molodensky, Bursa-Wolf and Molodensky-Badekas, which are described in Chapter 2.

There are also *non-conformal* transformations which allow for an element of distortion. Examples are given in Chapters 2 and 7 and include:

- An affine transformation, which preserves the ratio of the lengths of parallel line segments. It does not necessarily preserve angles between lines or distances between points, but it does preserve ratios of distances between points lying on a straight line.
- A pair or trio of multiple regression equations which represent a component of a datum shift as a linear combination of simple functions (usually monomials).
- A composite method. Datum shifts at data points are treated as a trend model plus a residual shift, the latter to be determined by a surface-fitting technique (SFT). This can be made into a software tool to estimate datum shifts wherever they are required or it can be used as a one-off process to estimate datum shifts at regularly-spaced points to enable straightforward interpolation. Transformations that conclude with

interpolation in the local grid-square are sometimes referred to as "grid look-up" (see, for example, Greaves [2004], Chen [2005]). They include OSTN02 and OSTN15.

The variety of surface-fitting techniques, generally designed to interpolate data at irregularly-spaced points, means that composite methods alone cover a large number of transformation methods, each with its own distinctive characteristics. Choosing the best one to represent datum shifts is inevitably going to be a matter of trial-and-error with different results in different cases.

It follows that any study of datum transformations needs to cover a wide range of methods to do justice to the wide variety of practical situations.

1.7 Knowledge gap and problem statement

The underlying problem is fitting a function over a region, given a finite set of physical data which is itself prone to errors. This is particularly true of results from older survey methods that were heavily reliant on angle measurement and preceded some or all of the distance-measurement technology that has evolved in the last 60-70 years. The relative positions of points in a survey network are known less accurately from historic observations in an old datum than from more recent observations in a newer datum; hence the element of mismatch in configuration.

Applying a simple model that fits the data approximately, may fail to capture one or more distortions in the relationship between datums. Conformal transformations between OSGB36 and modern satellite-based datums are not capable of reflecting the scale variations in the northern parts of Great Britain.

Applying a sophisticated model that fits the data exactly may prove to be unrealistic. Measurement errors of opposite sign at points which are near to each other can lead to distortions in the model which do not reflect the physical reality.

Even a sophisticated model that is designed to fit data approximately can be a bad reflection of reality. In Ghana's Golden Triangle, the data for the Accra datum and WGS84 only covers the inner part of the region, so the extrapolation effect has to be considered which rules out several types of function.

The challenge is to try and model datum transformations in a way that balances these considerations whilst being computationally comprehensible.

1.8 Review of transformation methods

There are published overviews of basic transformation methods that aim to be comprehensive, but in general they are incomplete. For example, Rapp (1993) and Varga *et al* (2017) both omit transformations of grid coordinates, and Knippers (2009) omits truly conformal 3D transformations. The latter two sources also contain errors (noted in Chapters 2 and 3). Perhaps the most comprehensive overview of basic methods is Ruffhead and Whiting (2020) which corresponds closely to Sections 2.1 to 2.13 of this thesis.

Where transformation methods have been published, there is a notable tendency for different authors to quote different forms and not acknowledge that the alternative form exists. One example of this, with citations, is given in Section 2.6 with different versions of the Helmert (3D conformal) transformation. Another, again with citations, is given in Section 2.8 with the fully-linear and partially-linear versions of the Bursa-Wolf transformation

NGA (2014) recommends a wide range of Standard Molodensky transformations from local datums to WGS84, most of them published in 1987-1997. Although these transform geodetic coordinates directly (without conversions to and from Cartesian coordinates), they only approximate to conformal transformations.

Both the Standard and Abridged Molodensky Datum Transformation Formulae use parameters ΔX, ΔY and ΔZ. As noted in DMA (1987a), these are traditionally derived from the mean differences between Cartesian coordinates. Molnár and Timár (2005), however, derived ΔX, ΔY & ΔZ from the actual Abridged Molodensky formulae for latitude & longitude shifts.

Building on the work of Appelbaum (1982), multiple regression equations (MREs) are provided in NIMA (2000) and NGA (2014) as a method of transforming local-datum coordinates to WGS84 coordinates. However, some of Appelbaum's methodology has been applied too literally, and the unimaginative definition of intermediate variables (combined with insufficient thought to the rounding of coefficients) has a knock-on effect on accuracy. NGA doesn't always correctly state the regions for which the MREs are valid. Also, its stated method of using MREs in reverse does not reverse the transformation exactly.

MREs for datum transformations have also been published by Dawod and Alnaggar (2000), Gledan and Azzeidani (2014) and González-Matesanz *et al* (2003). As with NIMA (2000) and its sequel NGA (2014), none of them has considered variations on MREs to reduce the perceived need for high-order polynomials which tend to cause instability.

Other researchers have studied the reversibility of transformations in a rather piecemeal fashion. Iliffe and Lott (2008) appear to be alone in considering the reversibility of several transformations. However, their criterion is whether the same formula can be re-used in a particular way in the reverse direction.

The derivation of basic transformations depends on whether they are linear with respect to their parameters. Those that are linear have usually been determined by the least-squares optimisation formulae in the introduction to Chapter 4, which can also be found in Bomford (1980) and Cross (1983). Some authors prefer "total least-squares" involving corrections to observed coordinates; see, for example, Pan et al (2015) and Laari et al (2016).

Multiple regression equations of the type proposed by Appelbaum (1982) and NIMA (2000) go beyond least-squares by reviewing and revising the set of polynomials used in the transformation; after each least-squares optimisation, tests are performed to determine which terms are statistically significant. Section 7.2.4.3.3 of DMA (1987a) has a description of how this is done for the NIMA/NGA MREs, a "stepwise regression" process. The complexity of the method is reflected in the number of authors (eg Kutoglu [2009b], Mitsakaki *et al* [2006], Ayer *et al* [2010]) who opted for a predetermined set of polynomials and applied no statistical-significance tests.

Optimising non-linear transformations has been the subject of much research. Methods published by Awange, Paláncz and others are listed in Sections 5.1 and 5.4. They have adverse characteristics ranging from missing details and questionable results to techniques which are not easily understood.

As noted in Section 1.3, the position of a point in a local datum relative to a global coordinate system can change over time. As a result, a transformation between a local datum and datum realisations is dependent on time, making it a dynamic (as opposed to static) transformation.

A transformation method can be made dynamic by doubling the number of transformation parameters. Each parameter is replaced by its value at a particular epoch and its rate of change (which becomes a multiple of elapsed time). Examples of this approach can be found in Dawson and Woods (2010), NGA (2014, Section 7.3) and Rabah *et al* (2016).

Such a rigorous analysis of dynamic transformations is probably unnecessary for practical purposes. If static transformations are known between a local datum and realisations at two different epochs, the respective transformed coordinates can easily be interpolated with respect to time if an intermediate transformation is required. Consequently, this thesis concentrates on static transformations.

The application of surface-fitting techniques to datum transformations, combined with a detailed comparison, has been undertaken by Grgić *et al* (2016). They used 12 techniques available in software packages, notably Surfer. Their comparisons only apply to 2D datum shifts in Croatia and there are in any case mistakes (noted in Sections 7.7 and 7.10). They used a 7-parameter trend model (which required coordinate conversions to and from Cartesian coordinates) and converted coordinates to a projected grid for the purpose of piecewise bilinear interpolation.

The use of piecewise bilinear interpolation over rectangular grids appears to be standard for grid look-up transformation methods. There is an acknowledgement by Iliffe and Lott (2008, page 106) that bicubic interpolation would give smoother results, but they conclude that the "additional complexity" of using 16 surrounding points is not worth it.

Least-squares collocation is a surface-fitting technique used by Grgić *et al* (2016), Deakin *et al* (1994), El-Mewafi (2015) and others for approximating datum shifts. In general, they use a correlation model where even faraway points have an effect on interpolation, albeit a small one. In the course of this study, no instances were found of finite covariance functions being used in this application, despite the existence of such functions in Sansò and Schuh (1987).

Of the surface-fitting techniques in Surfer tested by Grgić, several had already been applied to datum transformations. Examples include:

- Kriging in South Africa (Merry and Whittal [1998]);
- rubber sheeting in Great Britain (Ordnance Survey [2018]) and in Slovenia (Berk and Komadina (2013]);

- minimum curvature in North America (Dewhurst [1990]).

All these examples except Slovenia are grid look-up transformations.

Inverse distance to a power is another surface-fitting technique used by Grgić *et al* (2016), and it also has the characteristic that faraway points affect interpolation.

Of the other techniques in Surfer (2002), Nearest Neighbour has the advantage of retaining data-point values when "gridding" (interpolating at mesh points). The disadvantage is discontinuities which lead to the final interpolated surface to alternate between too flat and too steep.

1.9 Novelty and contribution to knowledge

This thesis not only acknowledges the different versions of the Helmert transformation but establishes the relationship between them (Section 2.6). It does the same with the fully-linear and partially versions of Bursa-Wolf (Section 2.8). In both cases, practical applications are given: the inverse transformation of Helmert in Section 3.6) and the derivation of partially-linear Bursa-Wolf (Section 4.8).

This thesis describes the SMITSWAM method introduced in Ruffhead (2016). SMITSWAM converts geodetic coordinates directly but improves on Standard Molodensky by being fully conformal (Section 2.14).

The derivation of Abridged Molodensky parameters in Molnár and Timár (2005) was the inspiration for the new partially-conformal variations on the Abridged Molodensky formulae defined in Section 2.16. ΔX, ΔY and ΔZ are optimised in different ways for different components of the transformation: one trio of ΔX, ΔY and ΔZ for transforming (ϕ, λ), another trio of ΔX, ΔY and ΔZ for transforming height. This thesis also provides similar variations for Standard Molodensky formulae (Section 2.15). In both cases, the new formulae enable a closer fit to actual data, and applying the formulae involves no more computation than the traditional versions.

This thesis addresses the weaknesses in the MREs compiled in NIMA (2000) and NGA (2014). Section 3.13 advocates a method that reverses MRE transformations exactly. Proper normalisation of both intermediate variables is recommended in Section 2.17. That section also new types of MREs consisting of piecewise low-order polynomials which form a smooth

patchwork, thereby reducing the perceived need for high-order polynomials which tend to cause instability.

This thesis treats reversibility as the existence of a reverse procedure which is computationally accurate relative to the formula(e) of any particular forward transformation. Chapter 3 addresses this issue for all the basic transformation methods. The reverse transformations in Sections 3.4 to 3.6 and 3.13 to 3.17 are new. Others are derived from first principles and, apart from the reverse transformation of Bursa-Wolf (see paragraph on Aktuğ in Section 3.8), it is not known whether they exist in sources other than those researched for this thesis.

New methods have been found for optimising non-linear transformations. Sections 5.1 and 5.2 describe a four-stage algorithm to derive Helmert transformations (Sections 5.1 and 5.2); this is based on distance analysis to resolve scale followed by optimisation of a rigid transformation. Sections 5.3 and 5.4 describe algorithms to optimise affine transformations with 8 or 9 parameters; these also involve optimisation of rigid transformations, but scale is resolved by effective application of an "equivalent enlargement hypothesis".

Optimising the new types of MRE in Section 2.17 is very much an analogous process to optimising the traditional MREs described in Section 2.13. However, rather than attempt the complex "stepwise regression" to apply the criteria of statistical significance, Section 5.5 proposes the simpler "eliminating ratios less than one" (ERLTO), in which the absolute value of each derived coefficient is compared with its standard error.

The use of surface-fitting techniques by Grgić *et al* (2016) in composite transformation methods used was a major influence on this study. However, the analysis in this thesis differs, in that unnecessary coordinate conversions are avoided and a smoother alternative to piecewise-bilinear interpolation is applied in the final stage. The full list of differences is:

- The trend model consists of bilinear functions of latitude and longitude rather than a 7-parameter conformal model, thereby removing the need to convert to and from Cartesian coordinates.
- The surface-fitting techniques are applied to geodetic coordinates (ϕ, λ) rather than projected grid coordinates.
- Original software was used for surface-fitting, although account was taken of method descriptions in Surfer (2002).

- Bicubic as well as bilinear interpolation was applied to the regularly-spaced datum shifts computed by the composite methods.

As an alternative to bilinear interpolation, the use of piecewise bicubic interpolation over rectangular grids is examined in Section 6.3. It is shown that only 12 surrounding points (rather than 16) are required, and modifications are provided when not all 12 are available. Three possible implementations are proposed, one of which simplifies interpolation by use of a surrounding corridor.

Least-squares collocation was studied as a surface-fitting technique with the signal representing the datum shift after the removal of a "trend" model. To eliminate the influence of faraway points on interpolation, two Sansò-and-Schuh finite covariance functions were applied to this type of signal, as was the more usual Gaussian function (Section 7.1). All types were analysed in the case studies, in which the selected trend model was bilinear.

Section 7.13 introduces a new variation to inverse-distance-to-a power interpolation, namely Hybrid Inverse Power Function Embodying Accelerated Decline (HIPFEAD). This introduces a limit-of-influence, thereby increasing the weight of nearby points in the interpolation process. The subtype Inverse Square Function Embodying Accelerated Decline (HISFEAD) also has potential use as a localised supplement to other methods to prevent exaggerated distortion (Sections 7.13 and 8.3).

Section 7.14 introduces the new Linear Interpolation Variation On Nearest Neighbour (LIVONN). This shares the Nearest-Neighbour method's advantage of retaining data-point values when "gridding". However, LIVONN replaces the discontinuities of Nearest Neighbour with transitions which bring interpolation closer to reality.

Chapter 7 explores the possibility of re-using a surface-fitting model whenever new estimates of datum shifts are required, instead of relying on grid look-up at regularly-spaced computation points (which is approximating from approximations). To this end, the computation and storage of a "revamped signal vector" is proposed for least-squares collocation and radial basis functions, enabling a more direct approach without repeating the whole derivation process.

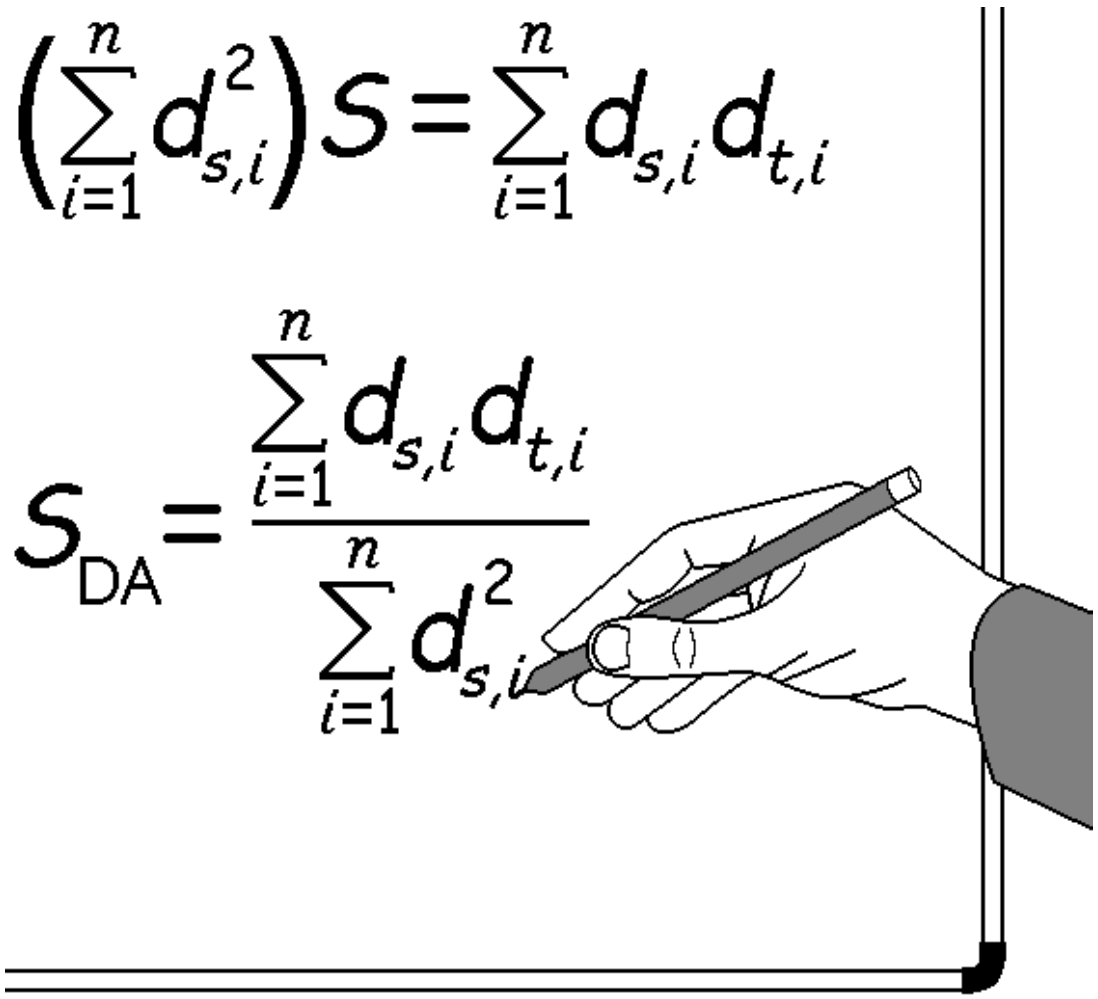
It is intended that these findings will be of value to the worldwide geoscientific community as it continues to develop and apply transformations between geodetic datums (including realisations).

With the exception of the LSS package used for contour maps, all the software was written by this researcher, using Excel Visual Basic For Applications. A selection of key macros and subroutines is described in Appendix G.

Part Two: Methods

$$\left(\sum_{i=1}^{n} d_{s,i}^{2}\right) S = \sum_{i=1}^{n} d_{s,i} d_{t,i}$$

$$S_{DA} = \frac{\sum_{i=1}^{n} d_{s,i} d_{t,i}}{\sum_{i=1}^{n} d_{s,i}^{2}}$$



CHAPTER 2: BASIC TRANSFORMATION METHODS

Geodetic datum transformations are generally attempts to model the changes that are evident from the differing coordinates of common points. Very often, they are between a local and global datum, but they can also be between two local datums. The subscripts *s* and *t* are used to indicate source datum and target datum respectively. The symbol Δ, as in $\Delta\phi$, denotes the "shift" which is added to the source coordinate to obtain the target coordinate.

Transformations can usually be placed into one of the following categories:

- An individual method that is relatively easy to apply and has its own distinctive name, such as a conformal transformation with 3 (or 7) parameters.
- A transformation-enhancement method which is designed to eliminate or reduce the residuals from an individual method; it would be a method in its own right if the identity function was treated as the individual method.
- A composite method which is an individual transformation followed by one or more enhancement methods.

The methods in this chapter are the first type, and are generally meant to fit data points in a least-squares sense rather than exactly. They can be used:

- for representing a transformation for which there is little or no distortion;
- for representing transformations where the accuracy requirement is not too stringent;
- to provide a "trend" model which leaves residuals to be fitted in some other manner.

Most methods in this chapter involve up to 12 parameters with geometrical meaning. The exception is multiple regression equations; these are intended to capture distortion more effectively than a conformal or affine transformation, and accuracy-of-fit is largely determined by the number of terms. MREs with a relatively small number of terms can be used as a trend model in the sense described above. As such they were considered by Varga *et al* (2017).

Sections 2.1 to 2.13 describe known methods. Sections 2.14 to 2.17 describe new methods.

The list of transformations in Varga *et al* (2017) was a record of those used in a particular case study involving Croatia. Nevertheless, it came close to providing a comprehensive list of basic methods. The methods covered in Sections 2.1 to 2.13 match that list apart from:

- The inclusion of 2-dimensional conformal and affine transformations.

- The exclusion of composite methods designed to correct distortions in datum transformations; these are considered in Chapter 7.

In this chapter, each transformation method is described in terms of the coordinates to which it is actually applied. Where those coordinates are Cartesian, it means that

- geodetic coordinates (ϕ, λ, h) in the source datum need to be converted to Cartesian coordinates (X, Y, Z) before transformation, and
- the transformed Cartesian coordinates need to be converted to geodetic coordinates in the target datum.

From equations (1-2) to (1-4), it is clear that the first of those stages is problematic if the source datum does not provide ellipsoidal heights. If that datum is locally defined, there are three possibilities:

- Neither orthometric nor ellipsoidal heights are known.
- Orthometric heights are available from levelling, but there is no local information on the separation between ellipsoid and geoid.
- Orthometric heights are available from levelling and a geoid model is available to convert them into ellipsoidal heights, although they are likely to be poor-quality approximations compared to ellipsoidal heights obtained from GNSS.

The datum transformations in this chapter are two-dimensional or three-dimensional. However, it should be noted that sometimes a three-dimensional method is used to obtain a two-dimensional result, making it a "2D-via-3D" transformation. If a three-dimensional model was obtained from data with ellipsoidal-height differences of poor quality, it will – at best - only be suitable for approximating horizontal datum shifts. When 2D-via-3D transformations are performed, the initial ellipsoidal heights are usually set to zero if no approximate values are available. The effect of doubtful or poor-quality height values on latitude and longitude is quite small, at least on the evidence of examples considered in Sections 2.3.

The properties of the transformations are given, in particular whether they are conformal, near-conformal or non-conformal. Conformal models, alternatively known as similarity transformations, preserve shape; a special case is a *rigid* transformation that preserves shape and size. Near-conformal models fall short of full conformality, usually because of simplifications in the formulae used. A special case of the non-conformal type is the affine

transformation, which preserves collinearity along a straight line; it allows for changes in angles and lengths, so there can be different scale changes in different directions.

Each method described in this Chapter is treated as a direct transformation of the type shown in Figure 1-7. Even where it commonly used as the central stage of a multi-stage transformation (Figure 1-8), the assumption is that the coordinate conversions are covered in Chapter 1.

2.1 Conformal transformation in 2 dimensions

A conformal grid-to-grid transformation has 4 parameters: two translations, one scale factor and one rotation. The model is applied to grid coordinates.

For this type of transformation to give satisfactory results, there needs to be a strong resemblance between the projections which generate the grids. In particular, the variations in scale factor across the area of interest must be proportional to each other. The dataset in sub-appendix C.7.7 is an example that meets this requirement. This issue is discussed in more detail in Iliffe and Lott (2008, 4.5.2).

The position-vector rotation convention, shown in Figure 2-1, is the one adopted here. Some authors prefer the coordinate-frame rotation convention shown in Figure 2-2 which has the opposite effect.

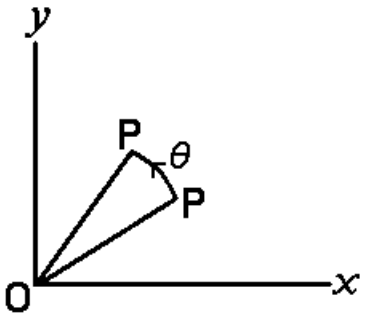


Figure 2-1: Rotation convention for a position vector in the O*xy* plane when the axes are regarded as fixed.

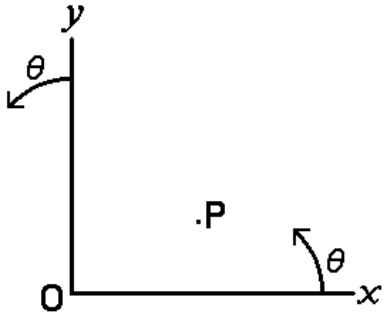


Figure 2-2: Rotation convention for the coordinate frame O*xy* when P is regarded as fixed.

The transformation equation is

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} + S \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_s \\ y_s \end{bmatrix}. \quad (2\text{-}1)$$

This is exactly equivalent to

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} + \begin{bmatrix} a_1 & -a_2 \\ a_2 & a_1 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \end{bmatrix} \quad (2\text{-}2)$$

so that Δx, Δy, a_1 and a_2 form an equivalent set of parameters to Δx, Δy, S and θ.

Equation (2-2) can be rewritten in the following form, which is linear with respect to Δx, Δy, a_1 and a_2.

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_s & -y_s \\ 0 & 1 & y_s & x_s \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ a_1 \\ a_2 \end{bmatrix}. \tag{2-3}$$

An example of a conformal grid transformation is shown in Figure 2-3. Using five points that form the vertices of a polygon, it illustrates a translation, scale change and rotation. In this case the rotation is negative.

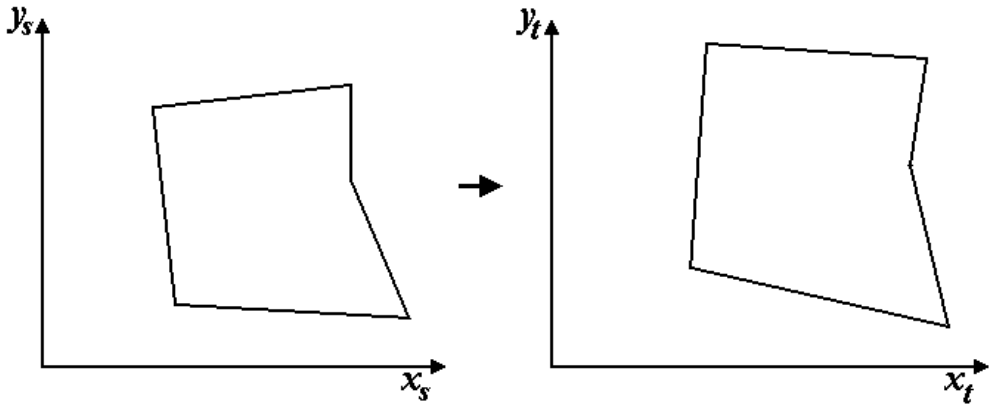


Figure 2-3: Illustration of a conformal grid transformation.

If the rotation convention is opposite to that shown in Figure 2-1, then each of the square matrices in equations (2-1) and (2-2) is replaced by its transpose, as is the submatrix of x_s and y_s terms in (2-3).

2.2 Affine transformation in 2 dimensions

An affine grid-to-grid transformation has 6 parameters: $\Delta x, \Delta y, a_1, a_2, a_3, a_4$. The model is applied to grid coordinates, and there needs to be a strong resemblance between the projections which generate the grids (as there is for the dataset in sub-appendix C.7.7). The basic equation is

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} + \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \end{bmatrix}. \tag{2-4}$$

This can be rewritten as

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_s & y_s & 0 & 0 \\ 0 & 1 & 0 & 0 & x_s & y_s \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \tag{2-5}$$

2.3 Rigorous 3-parameter conformal transformation

It is customary to reserve the description "3-parameter conformal transformation" for the transformation with three *translation* components, ΔX, ΔY and ΔZ (or T_X, T_Y and T_Z). The transformation of WGS72 to WGS84 falls outside this category, as it involves 3 parameters of different type (one translation, one rotation and one scale change).

In this paper, the method is called "3PC" (sometimes prefaced by "rigorous" as a reminder). It is applied to geocentric Cartesian coordinates (X, Y, Z). The basic equation is

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix}. \tag{2-6}$$

The Cartesian coordinates may have been obtained from geodetic coordinates where ellipsoidal heights are either dummy values or poor estimates. Full 3-stage computations were carried out for this study based on 4 transformations in NIMA (2000). Table 2-1 shows how ϕ_t and λ_t were affected by a 500m change in ellipsoidal height. The effect is small and is proportionately smaller for lesser changes in h. This demonstrates that ϕ_t and λ_t from the 3PC transformation are relatively insensitive to uncertainty about h_s.

Table 2-1: Effect of ellipsoidal-height changes (0→500m) on transformed latitude and longitude, using ΔX, ΔY, ΔZ shifts published in NIMA (2000) and NGA (2014)

Country covered by points shown in Figure 2-4	Datum transformation applied at h = 0 and h = 500m	RMS Effect on Latitude	RMS Effect on Longitude
Great Britain (35 points)	OSGB36→WGS84	0.0031m	0.0073m
Australia (75 points)	AGD84→WGS84	0.0125m	0.0101m
Congo (54 points)	Arc 1950→WGS84	0.0199m	0.0040m
India (81 points)	Indian→WGS84	0.0065m	0.0146m

Iliffe and Lott (2008, Section 4.3.2) summarise the limitations and merits of 3PC as follows:

> "For most classically defined and realised coordinate reference systems, one would expect errors of perhaps 5 to 10 m or more to result when using such a three-parameter transformation to convert GPS-derived data into a local mapping system. It has the advantage of being a compact transformation, however, and it is how simple hand-held GPS devices store and apply the details of transformations to many national mapping coordinate reference systems."

2.4 Standard Molodensky transformation

This method is documented in Molodensky *et al* (1962). It is a close approximation to the 3-parameter conformal transformation, so it is best classified as near-conformal. Its main attraction is that it is applied to geodetic coordinates.

The equations used by Standard Molodensky are truncated series expansions. A detailed derivation can be found in Deakin (2004). For some purposes the reduced accuracy is acceptable because of relative simplicity; conversion to and from Cartesian coordinates is not required. It is recommended in NIMA (2000) and NGA (2014). $\Delta\phi$ and $\Delta\lambda$ are given below in radians, but can easily be converted to arc-seconds or degrees.

$$\begin{aligned}\Delta\phi = [&-\Delta X \sin\phi_s \cos\lambda_s - \Delta Y \sin\phi_s \sin\lambda_s \\ &+\Delta Z \cos\phi_s + \Delta a(\nu_s e_s^2 \sin\phi_s \cos\phi_s)/a_s \\ &+\Delta f(\rho_s a_s/b_s + \nu_s b_s/a_s) \sin\phi_s \cos\phi_s]/(\rho_s + h_s)\end{aligned} \tag{2-7}$$

where

$$\rho_s = \frac{a_s(1-e_s^2)}{(1-e_s^2 \sin^2\phi_s)^{3/2}}\,. \tag{2-8}$$

$$\Delta\lambda = \frac{-\Delta X \sin\lambda_s + \Delta Y \cos\lambda_s}{(\nu_s + h_s)\cos\phi_s} \tag{2-9}$$

where ν_s is given by (1-1) applied in the source datum.

$$\begin{aligned}\Delta h = &\Delta X \cos\phi_s \cos\lambda_s + \Delta Y \cos\phi_s \sin\lambda_s \\ &+\Delta Z \sin\phi_s - \Delta a(a_s/\nu_s) + \Delta f(b_s/a_s)\nu_s \sin^2\phi_s.\end{aligned} \tag{2-10}$$

Kinneen and Featherstone (2004) notes that "depending on one's viewpoint", Standard Molodensky can be regarded as having three or five parameters. Some geodesists treat Δa and Δf as 4th and 5th parameters. However, these quantities are differences between ellipsoid-defining constants, so are unlike ΔX, ΔY and ΔZ. There are only 3 parameters in the 3PC method that (2-7)-(2-9) approximate. Only ΔX, ΔY and ΔZ (or T_X, T_Y and T_Z) are determined by an optimisation process from common data points. Accordingly, Standard Molodensky is regarded as a 3-parameter transformation in this study.

For the purpose of analysing Standard Molodensky – and similar methods – four transformations whose parameters are given in NIMA (2000) were selected, for both Ruffhead (2016) and this study:

- Ordnance Survey of Great Britain 1936 (OSGB36) to WGS84 over Great Britain;
- Australian Geodetic Datum 1984 (AGD84) to WGS84 over Australia;
- Arc 1950 (Zaire) to WGS84 over Congo;

- Indian Datum to WGS84 over India and Nepal.

The test points chosen for this study are shown in Figure 2-4. They differ from those used in Ruffhead (2016) in that this time they are aligned more precisely with the areas they were designed to cover. The revised computations for this study show that the Standard Molodensky approximations agree with the conformal model to within maximum differences of 0.027m in latitude, 0.047m in longitude, 0.016m in height and 0.050m in 3D distance. The RMS of the 3D distance differences is 0.019m.

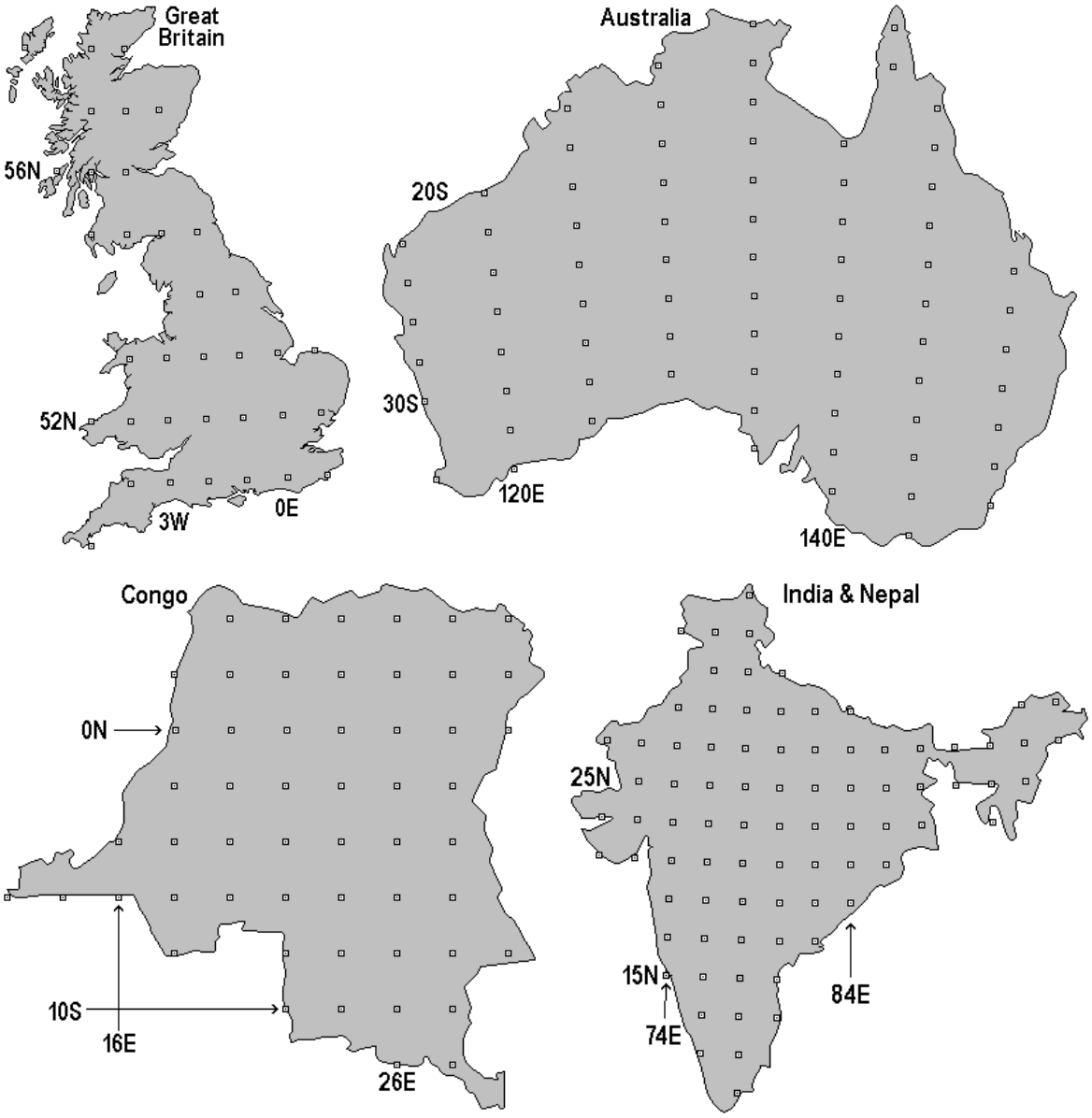


Figure 2-4: Test points selected for the new study of NIMA/NGA Standard Molodensky transformations between local datums (OSGB36, AGD84, Arc 1950, Indian) and WGS84. Ellipsoidal heights used were 0, 500m, 2km and 10km.

Equations (2-7) and (2-8) both depend on ellipsoidal heights, and sometimes these are either dummy values or poor estimates. Standard Molodensky transformations were compared between ellipsoidal heights 0 and 500m for the points in Figure 2-4. The results in Table 2-1 were duplicated completely to the precision shown. The effect is small and is proportionately smaller for lesser changes in h. This demonstrates that ϕ_t and λ_t from Standard Molodensky are relatively insensitive to uncertainty about h_s.

2.5 Abridged Molodensky transformation

This is a close approximation to the 3-parameter conformal transformation, although less so than Standard Molodensky. It is best classified as near-conformal. Its main attraction is that it is applied to geodetic coordinates.

The "Abridged" version had been selected by DMA for use with WGS66. DMA (1987a, section 7.2.4.3.2) noted that "the Abridged Molodensky Datum Transformation Formulas have been used more extensively that the Standard Formulas", but on grounds of accuracy firmly recommended the latter in "any new software involving the Molodensky Datum Transformation Formulas". Accordingly, DMA (1991) and its successors NIMA (2000) & NGA (2014) do not mention Abridged Molodensky. Nevertheless, it continues to be used. As pointed out by Iliffe and Lott (2008, Section 4.4.2) the differences between Abridged and Standard are not significant "in the context of a 3-parameter transformation, which has a typical accuracy in the order of 5 to 10 m".

The equations do not use the ellipsoidal heights and it is not uncommon for the 3rd equation to go unused. A detailed derivation can be found in Deakin (2004). $\Delta\phi$ and $\Delta\lambda$ are given below in radians, but can easily be converted to arc-seconds or degrees.

$$\Delta\phi = [-\Delta X \sin\phi_s \cos\lambda_s - \Delta Y \sin\phi_s \sin\lambda_s + \Delta Z \cos\phi_s + (a_s \Delta f + f_s \Delta a) \sin 2\phi_s]/\rho_s. \quad (2\text{-}11)$$

$$\Delta\lambda = \frac{-\Delta X \sin\lambda_s + \Delta Y \cos\lambda_s}{\nu_s \cos\phi_s}. \quad (2\text{-}12)$$

$$\Delta h = \Delta X \cos\phi_s \cos\lambda_s + \Delta Y \cos\phi_s \sin\lambda_s + \Delta Z \sin\phi_s + (a_s \Delta f + f_s \Delta a) \sin^2\phi_s - \Delta a. \quad (2\text{-}13)$$

The radii of curvature ν and ρ are given by (1-1) and (2-8) respectively.

The test points chosen for this study are the same as those shown in Figure 2-4. (The differences from those used in Ruffhead (2016) are explained in Section 2.4.) The revised computations for

this study show that the Abridged Molodensky approximations agree with the conformal model to within maximum differences of 0.687m for latitude, 0.627m for longitude, 0.125m for height and 0.735m for 3D distance. The RMS of the 3D distance differences is 0.125m.

For points on the ellipsoid, the longitude error is the same as that for Standard Molodensky, which is to be expected from a quick comparison of (2-12) with (2-9). Longitude error for Abridged Molodensky increases as ellipsoidal height increases. The latter property is also true of latitude error.

The case for regarding this method as having three parameters rather than five is the same as that given for Standard Molodensky in Section 2.4.

2.6 Rigorous 7-parameter conformal transformation

In addition to the 3 translation parameters, this method applies 3 rotations (R_X, R_Y and R_Z) and a scale change ΔS. It is the most general form of the 3-dimensional conformal transformation, unless local Cartesian coordinates are built into the method. It is applied to geocentric Cartesian coordinates (X, Y, Z).

For rotations to be unambiguous, a sign convention needs to be specified. There are two possibilities:

- Position vector (PV) rotations, describing rotation of position vectors about Cartesian axes;
- Coordinate frame (CF) rotations, describing rotation of Cartesian axes when points are considered as fixed.

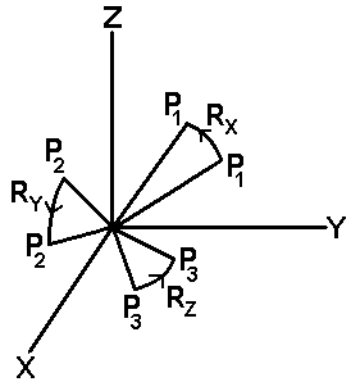


Figure 2-5: Rotation conventions for position vectors in the OYZ plane, the OZX plane and the OXY plane.

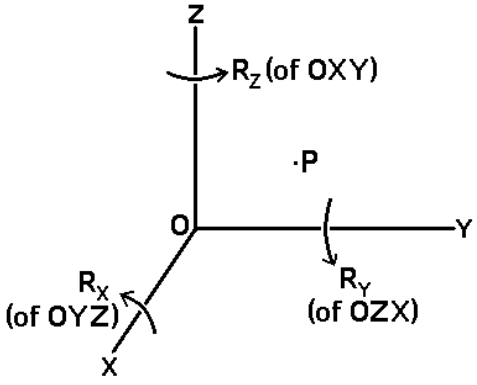


Figure 2-6: Rotation conventions for the coordinate frame when P is regarded as fixed.

Position vector rotations are illustrated in Figure 2-5. The sign convention adopted here is that positive rotations are counter-clockwise about Cartesian axis when viewed from the positive side of the origin, as in Figure 2-5. One characteristic is that a positive rotation about the Z-axis has the effect of increasing longitude. Ordnance Survey (2018, 37) says of PV rotations, "It is the form in most common use in Europe (particularly in the oil and gas industry), is used by the International Association of Geodesy (IAG) and recommended by ISO (2007) and is EPSG dataset coordinate operation method code 1033". This is the convention adopted for this thesis.

An alternative school of thought prefers to use coordinate frame rotations. According to this, the sign convention is that positive rotations of the axis-planes are counter-clockwise when viewed from the positive side of the origin, as in Figure 2-6. CF rotation parameters are opposite in sign to those of PV. Ordnance Survey (2018, 37) says the CF convention "is common in the USA oil and gas industry and is EPSG dataset coordinate operation method code 1032".

The rigorous 7PC method is referred to as the Helmert transformation in a number of sources, including Fan (2005), Hofmann-Wellenhof and Moritz (2006), Sjöberg (2013) and Watson (2005). However, the method is so often used in a simplified form, described below in Section 2.8, that the simplified form is often called a Helmert transformation. See, for example, NATO (2001), Zgonc (2006), Knippers (2009) and Andrei (2006). This paper only uses "Helmert" for the rigorous 7PC transformation, but also uses "rigorous 7PC" where there is a possible danger of ambiguity.

The generic 7PC equation is

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + (1 + \Delta S)\mathbf{R} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} \tag{2-14}$$

where **R** is a rotation matrix.

The precise form of **R** depends on the order in which the rotations R_X, R_Y and R_Z are applied. As Harvey (1986, page 107) observes: "In general, successive rotations of a body about fixed axes are not commutative. Thus the order of the rotations is important unless the rotations are small." In theory there are six possible permutations, but available evidence indicates that only two are used in practice.

- One permutation is where R_X is applied to the position vector first and R_Z last. This is the permutation described by Harvey (1986, page 107) as "the most commonly applied". It is the permutation used in Deakin (2006), Dewitt (1996), Fan (2005), Harvey (1986), Hofmann-Wellenhof and Moritz (2006), Reit (1998), Varga *et al* (2017), Watson (2005), Wolf and Ghilani (1997). For the purposes of this study, and in compliance with Ruffhead (2021a) and Ruffhead (2021b), this permutation is Version 1. Accordingly, Helmert Version 1 is the 7-parameter conformal transformation that uses this permutation of rotations.
- The other permutation is where R_Z is applied to the position vector first and R_X last, as in Awange and Grafarend (2002), Sjoberg (2013) and Wang *et al* (2018). For the purposes of this study, and in compliance with Ruffhead (2021a) and Ruffhead (2021b), this permutation is Version 2. Accordingly, Helmert Version 2 is the 7-parameter conformal transformation that uses this permutation of rotations.

In the case of Version 1, **R** is given by

$$\mathbf{R}_{ZYX} = \begin{bmatrix} c_Z & -s_Z & 0 \\ s_Z & c_Z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_Y & 0 & s_Y \\ 0 & 1 & 0 \\ -s_Y & 0 & c_Y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_X & -s_X \\ 0 & s_X & c_X \end{bmatrix}. \tag{2-15}$$

Following the shorthand notation used in Deakin (2006), c_X denotes $\cos R_X$, s_X denotes $\sin R_X$, etc. It is easily verified that

$$\mathbf{R}_{ZYX} = \begin{bmatrix} c_Y c_Z & s_X s_Y c_Z - c_X s_Z & s_X s_Z + c_X s_Y c_Z \\ c_Y s_Z & c_X c_Z + s_X s_Y s_Z & c_X s_Y s_Z - s_X c_Z \\ -s_Y & s_X c_Y & c_X c_Y \end{bmatrix}. \tag{2-16}$$

In the case of Version 2, **R** is given by

$$\mathbf{R}_{XYZ} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_X & -s_X \\ 0 & s_X & c_X \end{bmatrix} \begin{bmatrix} c_Y & 0 & s_Y \\ 0 & 1 & 0 \\ -s_Y & 0 & c_Y \end{bmatrix} \begin{bmatrix} c_Z & -s_Z & 0 \\ s_Z & c_Z & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2-17}$$

where c_X denotes $\cos R_X$, s_X denotes $\sin R_X$, etc. It is easily verified that

$$\mathbf{R}_{XYZ} = \begin{bmatrix} c_Y c_Z & -c_Y s_Z & s_Y \\ s_X s_Y c_Z + c_X s_Z & c_X c_Z - s_X s_Y s_Z & -s_X c_Y \\ s_X s_Z - c_X s_Y c_Z & c_X s_Y s_Z + s_X c_Z & c_X c_Y \end{bmatrix} \tag{2-18}$$

The sine terms in equations (2-15) to (2-18) change sign if the rotation convention is CF (as in Figure 2-6) rather than PV.

As an aside, the rotation matrices in (2-16) and (2-18) both have an interesting property. If R_X, R_Y and R_Z are replaced by $R_X \pm 180°$, $\pm 180° - R_Y$, and $R_Z \pm 180°$, the rotation matrix is completely unchanged. This is academic for geodetic datum transformations where rotations are quoted in arc-seconds rather than degrees. However, it has potential significance in other

areas of Geomatics, where rotations can be larger than 90º. Fan (2005, page 5) mentions the equivalence property for Helmert Version 1.

Special cases of the rigorous 7PC transformation include:

- The 4-parameter conformal model based on ΔX, ΔY, ΔZ and scale factor S; this is actually linear with respect to the parameters.
- The 4-parameter rigid transformation based on ΔX, ΔY, ΔZ and rotation R_Z.
- The 5-parameter conformal model based on ΔX, ΔY, ΔZ, rotation R_Z and scale factor S.
- The 6-parameter rigid transformation based on ΔX, ΔY, ΔZ and rotations R_X, R_Y and R_Z.

The equations of the 7-parameter method are easily modified for these cases.

Prior to Ruffhead (2021b), one issue that appears to have been unexplored is the possible equivalence of rotation-parameter sets between Versions 1 and 2. If rotation parameters have been derived for use in one version, is there an equivalent set which has the same effect when used in the other version? If so, one obvious practical application is where software is based on the opposite version to the version for which the rotation parameters are known.

As a working hypothesis, it was assumed that any given orientation of a rigid body in three-dimensional space can be achieved by rotations about fixed axes in any order. (This does not mean that the values of those rotations are independent of the order in which they applied.) The matter of formal mathematical proof will be addressed later.

It follows that if the rotation matrix is derived by either version of Helmert, there exists an alternative set of rotations such that the rotation matrix derived by the other version is the same. The following Helmert conversion algorithms demonstrate how those alternative rotations are computed. They have already appeared in Ruffhead (2021b).

2.6.1 Rotation-parameter conversion from Version 1 to Version 2

Since two sets of rotations are involved, R_X, R_Y, R_Z denote the original rotations and R'_X, R'_Y, R'_Z denote the converted rotations. In this case, the initial Version-1 rotations R_X, R_Y and R_Z are applied as per (2-15). By (2-16),

$$\begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix} = \begin{bmatrix} c_Y c_Z & s_X s_Y c_Z - c_X s_Z & s_X s_Z + c_X s_Y c_Z \\ c_Y s_Z & c_X c_Z + s_X s_Y s_Z & c_X s_Y s_Z - s_X c_Z \\ -s_Y & s_X c_Y & c_X c_Y \end{bmatrix}. \quad (2\text{-}19)$$

Bearing in mind (2-18), the first objective is to find $c'_X, s'_X, c'_Y, s'_Y, c'_Z, s'_Z$ such that

$$\begin{bmatrix} c'_Y c'_Z & -c'_Y s'_Z & s'_Y \\ s'_X s'_Y c'_Z + c'_X s'_Z & c'_X c'_Z - s'_X s'_Y s'_Z & -s'_X c'_Y \\ s'_X s'_Z - c'_X s'_Y c'_Z & c'_X s'_Y s'_Z + s'_X c'_Z & c'_X c'_Y \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix}. \quad (2\text{-}20)$$

Equating the expressions for $r_{1,3}$,

$$s'_Y = s_X s_Z + c_X s_Y c_Z. \quad (2\text{-}21)$$

On the basis that s'_Y is the sine of R'_Y,

$$(c'_Y)^2 = 1 - (s'_Y)^2 = 1 - (s_X s_Z + c_X s_Y c_Z)^2. \quad (2\text{-}22)$$

Making the assumption that $|R'_Y| \leq 90°$, which will always be the case for geodetic datum transformations,

$$c'_Y = \sqrt{1 - (s'_Y)^2}. \quad (2\text{-}23)$$

Equating the expressions for $r_{3,3}, r_{2,3}, r_{1,1}, r_{1,2}$ respectively,

$$c'_X c'_Y = c_X c_Y, \quad (2\text{-}24)$$

$$s'_X c'_Y = s_X c_Z - c_X s_Y s_Z, \quad (2\text{-}25)$$

$$c'_Y c'_Z = c_Y c_Z, \quad (2\text{-}26)$$

$$c'_Y s'_Z = c_X s_Z - s_X s_Y c_Z. \quad (2\text{-}27)$$

Since c'_Y is positive, the Version-2 rotations can be computed from

$$R'_X = \arctan2[c'_X c'_Y, s'_X c'_Y], \; R'_Y = \arctan2[c'_Y, s'_Y], \; R'_Z = \arctan2[c'_Z c'_Y, s'_Z c'_Y]. \quad (2\text{-}28)$$

The arctan2[x,y] function is arctan(y/x) in the range -180° to 180° such that it always has the same sign as x. Programming languages usually have a function corresponding to arctan2. In VBA, a user-defined version is needed; suitable code can be found, for example, in Ruffhead (2016). The use of arctan2 rather than arctan is not strictly necessary when rotations are known to be small, but its presence here is to enable generalisation.

While (2-21) to (2-28) are necessary to establish an equivalent set of Version-2 rotations, they are not sufficient without a mathematical proof of the identities in $r_{2,1}, r_{2,2}, r_{3,1}, r_{3,2}$ arising from (2-20). This is given in Appendix A.

When Helmert transformations are used in the wider field of geomatics, the rotations are not necessarily numerically smaller than 90º. It does no harm to compute c'_Y from (2-23), making it positive, making R'_Y numerically smaller than 90º, and enabling the application of (2-28). After all, there is an equivalent set of Version-2 rotations which makes R'_Y larger than 90º.

The only problem occurs when $c'_Y = 0$ because (2-28) will leave R'_X and R'_Z undefined. From (2-24) and (2-26), this scenario only occurs if $c_Y = 0$ or $c_X = c_Z = 0$. This in turn only occurs if R_Y is an odd multiple of 90º or R_X and R_Z are both odd multiples of 90º. Extensive manipulation of trigonometrical identities enabled the compilation of Table 2-2.

Table 2-2: Special cases of Helmert Version 1 → Helmert Version 2 rotation conversions

Helmert Version-1 rotations			Helmert Version-2 rotations (not necessarily unique)			Selected trigonometrical identities that assist the proof that the rotation matrices are the same
R_X	R_Y	R_Z	R'_X	R'_Y	R'_Z	
90º	R_Y	90º	90º	90º	$-R_Y$	
90º	R_Y	-90º	R_Y	-90º	-90º	
-90º	R_Y	90º	-90º	-90º	R_Y	
-90º	R_Y	-90º	R_Y	90º	-90º	
R_X	90º	R_Z	-90º	$R_X - R_Z$ $+90°$	90º	$\sin(R_X - R_Z + 90°) = \cos(R_X - R_Z) = c_X c_Z + s_X s_Z$; $\cos(R_X - R_Z + 90°) = -\sin(R_X - R_Z) = c_X s_Z - s_X c_Z$.
R_X	-90º	R_Z	90º	$R_X + R_Z$ $-90°$	90º	$\sin(R_X + R_Z - 90°) = -\cos(R_X + R_Z) = s_X s_Z - c_X c_Z$; $\cos(R_X + R_Z - 90°) = \sin(R_X + R_Z) = s_X c_Z + c_X s_Z$.

2.6.2 Rotation-parameter conversion from Version 2 to Version 1

Since two sets of rotations are involved, R_X, R_Y, R_Z denote the original rotations and R'_X, R'_Y, R'_Z denote the converted rotations. In this case, the initial Version-2 rotations R_X, R_Y and R_Z are applied as per (2-17). By (2-18),

$$\begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix} = \begin{bmatrix} c_Y c_Z & -c_Y s_Z & s_Y \\ s_X s_Y c_Z + c_X s_Z & c_X c_Z - s_X s_Y s_Z & -s_X c_Y \\ s_X s_Z - c_X s_Y c_Z & c_X s_Y s_Z + s_X c_Z & c_X c_Y \end{bmatrix}. \tag{2-29}$$

Bearing in mind (2-16), the first objective is to find c'_X, s'_X, c'_Y, s'_Y, c'_Z, s'_Z such that

$$\begin{bmatrix} c'_Y c'_Z & s'_X s'_Y c'_Z - c'_X s'_Z & s'_X s'_Z + c'_X s'_Y c'_Z \\ c'_Y s'_Z & c'_X c'_Z + s'_X s'_Y s'_Z & c'_X s'_Y s'_Z - s'_X c'_Z \\ -s'_Y & s'_X c'_Y & c'_X c'_Y \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix}. \tag{2-30}$$

Equating the expressions for $r_{3,1}$,

$$s'_Y = c_X s_Y c_Z - s_X s_Z. \tag{2-31}$$

On the basis that s'_Y is the sine of R'_Y,

$$(c'_Y)^2 = 1 - (s'_Y)^2 = 1 - (c_X s_Y c_Z - s_X s_Z)^2. \tag{2-32}$$

Making the assumption that $|R'_Y| \leq 90°$, which will always be the case for geodetic datum transformations,

$$c'_Y = \sqrt{1 - (s'_Y)^2}. \tag{2-33}$$

Equating the expressions for $r_{3,3}$, $r_{3,2}$, $r_{1,1}$, $r_{2,1}$ respectively,

$$c'_X c'_Y = c_X c_Y, \tag{2-34}$$

$$c'_Y s'_X = c_X s_Y s_Z + s_X c_Z, \tag{2-35}$$

$$c'_Y c'_Z = c_Y c_Z, \tag{2-36}$$

$$c'_Y s'_Z = s_X s_Y c_Z + c_X s_Z. \tag{2-37}$$

Since c'_Y is positive, the Version-1 rotations can be computed from

$$R'_X = \arctan2[c'_X c'_Y, s'_X c'_Y],\ R'_Y = \arctan2[c'_Y, s'_Y],\ R'_Z = \arctan2[c'_Z c'_Y, s'_Z c'_Y]. \quad (2\text{-}38)$$

The comment about the use of the arctan2 function which follows (2-28) also applies here.

While (2-31) to (2-38) are necessary to establish an equivalent set of Version-1 rotations, they are not sufficient without a mathematical proof of the identities in $r_{1,2}, r_{1,3}, r_{2,2}, r_{2,3}$ arising from (2-30). This is given in Appendix A.

When Helmert transformations are used in the wider field of geomatics, the rotations are not necessarily numerically smaller than 90º. It does no harm to compute c'_Y from (2-33), making it positive, making R'_Y numerically smaller than 90º, and enabling the application of (2-38). After all, there is an equivalent set of Version-2 rotations which makes R'_Y larger than 90º.

The only problem occurs when $c'_Y = 0$ because (2-38) will leave values R'_X and R'_Z undefined. From (2-34) and (2-36), this scenario only occurs if $c_Y = 0$ or $c_X = c_Z = 0$. This in turn only occurs if R_Y is an odd multiple of 90º or R_X and R_Z are both odd multiples of 90º. Extensive manipulation of trigonometrical identities enabled the compilation of Table 2-3.

Table 2-3: Special cases of Helmert Version 2 → Helmert Version 1 rotation conversions

Helmert Version-2 rotations			Equivalent Helmert Version-1 rotations (not necessarily unique)			Selected trigonometrical identities that assist the proof that the rotation matrices are the same
R_X	R_Y	R_Z	R'_X	R'_Y	R'_Z	
90º	R_Y	90º	R_Y	-90º	90º	
90º	R_Y	-90º	R_Y	90º	-90º	
-90º	R_Y	90º	R_Y	90º	90º	
-90º	R_Y	-90º	-90º	-90º	$-R_Y$	
R_X	90º	R_Z	-90º	$R_X + R_Z$ +90°	-90º	$\sin(R_X + R_Z + 90°) = \cos(R_X + R_Z) = c_X c_Z - s_X s_Z$; $\cos(R_X + R_Z + 90°) = -\sin(R_X + R_Z) = -(s_X c_Z + c_X s_Z)$.
R_X	-90º	R_Z	90º	$R_X - R_Z$ − 90°	-90º	$\sin(R_X - R_Z - 90°) = -\cos(R_X - R_Z) = -(c_X c_Z + s_X s_Z)$; $\cos(R_X - R_Z - 90°) = \sin(R_X - R_Z) = s_X c_Z - c_X s_Z$.

2.7 Rigorous localised 7-parameter conformal transformation

This is a similar method except that the rotations are related to a local centroid (X_m, Y_m, Z_m). In effect, it is applied to local Cartesian coordinates. The rotation convention adopted here is shown in Figure 1-3 and discussed in Section 2.6.

The transformation equation takes the form

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} X_m \\ Y_m \\ Z_m \end{bmatrix} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + (1 + \Delta S)\mathbf{R} \begin{bmatrix} X_s - X_m \\ Y_s - Y_m \\ Z_s - Z_m \end{bmatrix} \quad (2\text{-}39)$$

where the rotation matrix **R** is given by (2-15) and (2-16) or by (2-17) and (2-18).

Perhaps the best name for this transformation is "localised 7PC". It is, however, an alternative implementation of the non-localised version. Equation (2-39) can easily be converted into the form (2-14), with a change to the values of the translation parameters but not to the scaling or rotations. Conversely, equation (2-14) can be converted into the form (2-39).

Special cases of this 7-parameter conformal transformation include:

- The 4-parameter conformal model based on ΔX, ΔY, ΔZ and scale factor S; this is actually linear with respect to the parameters.
- The 4-parameter rigid transformation based on ΔX, ΔY, ΔZ and rotation R_Z.
- The 5-parameter conformal model based on ΔX, ΔY, ΔZ, rotation R_Z and scale factor S.
- The 6-parameter rigid transformation based on ΔX, ΔY, ΔZ and rotations R_X, R_Y, R_Z.

The equations of the 7-parameter method are easily modified for these cases.

Mention should be made of 7-parameter conformal transformations from X_s', Y_s', Z_s' to X_t', Y_t', Z_t' when X', Y' and Z' are local level coordinates as described in subsection 1.4.4. This study treats the local level coordinates as east, north & up, and relates them to geocentric Cartesians by equations (1-24) & (1-25). Subsection 1.4.4 noted that Rapp (1993, page 68) and Leick (1995, page 480) treat the local level coordinates as south, east and up; the process of transforming these coordinates has come to be known as the Veis transformation.

2.8 Simplified 7-parameter conformal transformation (Bursa-Wolf)

The transformation associated with Bursa (1962) and Wolf (1963) is a simplified version of the 7PC transformation. It is applied to geocentric Cartesian coordinates (X, Y, Z). The fully-linear version of Bursa-Wolf, abbreviated here as FLBW, takes the form shown in (2-40) below. It matches the form quoted by Ge *et al*, (2013), Kashani (2006), McCarthy (1996) and Yusheng (2014), although not all use the PV rotation convention.

Bursa-Wolf is wrongly described as conformal by Soycan and Soycan (2008), Okwuashi and Eyoh (2012) and Dhungana and Lama (2018). The equation is based on (2-14) but assumptions are needed about ΔS, R_X, R_Y and R_Z being small, less than $2''$ in the case of the latter three. (This requirement is sometimes overlooked, as in Knippers [2009].) Treating the

rotations as being in radians, second-order terms such as $\Delta S \cdot R_X$ and $R_Y R_Z$ are deemed to be negligible and the transformation is regarded as near-conformal. The equation takes the form

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} 1+\Delta S & -R_Z & R_Y \\ R_Z & 1+\Delta S & -R_X \\ -R_Y & R_X & 1+\Delta S \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} \tag{2-40}$$

As the designation "fully-linear" suggests, this expression is linear with respect to the parameters ΔX, ΔY, ΔZ, ΔS, R_X, R_Y and R_Z. This becomes more obvious when (2-40) is rewritten in the following way.

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & X_s & 0 & Z_s & -Y_s \\ 0 & 1 & 0 & Y_s & -Z_s & 0 & X_s \\ 0 & 0 & 1 & Z_s & Y_s & -X_s & 0 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \Delta S \\ R_X \\ R_Y \\ R_Z \end{bmatrix} \tag{2-41}$$

Sets of parameters for Bursa-Wolf transformations can be found in

- ESRI (2012) which covers a wide range used in ArcGIS;
- NGA (2008) which covers many used by NATO.

If the rotation convention is CF (as in Figure 2-6) rather than PV, then each rotation in equation (2-40) needs a change of sign. The same applies to the coefficients of R_X, R_Y and R_Z in equation (2-41).

Bursa-Wolf is sometimes quoted in a partially-linear form in which only the rotation matrix has been linearised. The abbreviation used here is PLBW, and the transformation formula is

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + (1+\Delta S) \begin{bmatrix} 1 & -R_Z & R_Y \\ R_Z & 1 & -R_X \\ -R_Y & R_X & 1 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix}. \tag{2-42}$$

ESRI (2008), Deakin (2006), Gacoki and Aduol (2002) and Iliffe and Lott (2008) are among the sources which prefer this form. The methods used in the latter 3 sources to derive the PLBW parameters are described in Section 4.8.

If the rotations in (2-40) and (2-42) are given the superscripts FLBW and PLBW respectively, then a necessary and sufficient condition for the formulae to be equivalent is that

$$\begin{bmatrix} R_X^{FLBW} \\ R_Y^{FLBW} \\ R_Z^{FLBW} \end{bmatrix} = (1+\Delta S) \begin{bmatrix} R_X^{PLBW} \\ R_Y^{PLBW} \\ R_Z^{PLBW} \end{bmatrix}. \tag{2-43}$$

This means that the FLBW rotations are less accurate approximations to the true rotations than the PLBW rotations, in the sense that the scaling introduces an additional source of error.

Another consequence of the relationship between the two forms of Bursa-Wolf is that (2-43) can be used as a substitution to convert PLBW into a fully-linear form.

Rapp (1993, 61) quotes a study by Malys (1988) on the numerical impact of the small angle approximation used in (2-14). He compared the elements of that linearised rotation matrix with those of the rigorous rotation matrix given above in (2-16). He found that the disagreement between corresponding elements was at the level of 0.5×10^{-11} when the rotation angles were on the order of 1″; at the level of 0.5×10^{-10} when the angles were on the order of 3″; and at the level of 0.5×10^{-9} when the angles were on the order of 9″. It was noted that "an error of 0.5×10^{-9} propagates into a coordinate error on the order of 3mm". However, NGA (2008) shows that rotation angles larger than 9″ do occur occasionally.

Iliffe and Lott (2008, Section 4.3.3) observes that "The 7-parameter transformation methods are most commonly encountered when transforming data acquired in a modern system such as GPS to a national coordinate reference system... Errors in the order of one, two or more metres might be encountered, depending on the extent of the area covered and the age of the original survey." Bursa-Wolf is the version of the 7-parameter transformation that occurs most in software packages and (like 3PC) enables them to store & apply the details of transformations to many national mapping coordinate reference systems.

2.9 Simplified localised 7-parameter conformal transformation (Molodensky-Badekas)

The Molodensky-Badekas method is a simplified version of the rigorous localised 7PC transformation based on a local centroid (X_m, Y_m, Z_m). In effect, the transformation is applied to local Cartesian coordinates.

The transformation formula is based on (2-39) but assumptions are made about ΔS, R_X, R_Y and R_Z being small (less than 2″ in the case of the latter three). Treating the rotations as being in radians, second-order terms such as $\Delta S{\cdot}R_X$ and $R_Y R_Z$ are deemed to be negligible and the transformation is regarded as near-conformal. The fully-linear version of Molodensky-Badekas, abbreviated here as FLMB, takes the form shown in (2-44) below.

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} X_m \\ Y_m \\ Z_m \end{bmatrix} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} 1+\Delta S & -R_Z & R_Y \\ R_Z & 1+\Delta S & -R_X \\ -R_Y & R_X & 1+\Delta S \end{bmatrix} \begin{bmatrix} X_s - X_m \\ Y_s - Y_m \\ Z_s - Z_m \end{bmatrix}. \tag{2-44}$$

As the designation "fully-linear" suggests, this expression is linear with respect to the parameters ΔX, ΔY, ΔZ, ΔS, R_X, R_Y and R_Z. This becomes more obvious when (2-44) is rewritten in the following way.

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & X_s - X_m & 0 & Z_s - Z_m & -(Y_s - Y_m) \\ 0 & 1 & 0 & Y_s - Y_m & -(Z_s - Z_m) & 0 & X_s - X_m \\ 0 & 0 & 1 & Z_s - Z_m & Y_s - Y_m & -(X_s - X_m) & 0 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \Delta S \\ R_X \\ R_Y \\ R_Z \end{bmatrix}. \tag{2-45}$$

Section 4.1 of Varga *et al* (2017) says this is a 7-parameter method that some would call a 10-parameter transformation. Some authors refer to it as a "7+3" model. This distinguishes the 7 parameters derived by least-squares (translation, scale-change and rotations) from the 3 centroid coordinates (obtained by a simple formula).

Because of its use of coordinate differences, equation (2-45) avoids the large numbers that occur in equation (2-41). The contrast is even more marked in normal equations based on (2-45) and (2-41). The computations carried out with both methods during this study (with double precision) suggest that the large numbers are not a problem.

It can be argued that Molodensky-Badekas is an alternative implementation of Bursa-Wolf. Al Marzooqi *et al* (2005, Section 2.4.3) describes them as theoretically identical. Equation (2-44) can easily be converted into the form (2-40), although the values of the translation parameters will change. Conversely, equation (2-40) can be converted into the form (2-44). The relationship between the translation parameters arising from the two methods can easily be shown to be

$$\begin{bmatrix} \Delta X^{\mathrm{FLBW}} \\ \Delta Y^{\mathrm{FLBW}} \\ \Delta Z^{\mathrm{FLBW}} \end{bmatrix} = \begin{bmatrix} \Delta X^{\mathrm{FLMB}} \\ \Delta Y^{\mathrm{FLMB}} \\ \Delta Z^{\mathrm{FLMB}} \end{bmatrix} - \begin{bmatrix} \Delta S & -R_Z & R_Y \\ R_Z & \Delta S & -R_X \\ -R_Y & R_X & \Delta S \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \end{bmatrix}. \tag{2-46}$$

This has been noted in vector-matrix notation by Deakin (2006, page 20).

If the rotation convention is the opposite of that shown in Figure 2.4, then each rotation in equation (2-44) needs a change of sign. The same applies to the coefficients of R_X, R_Y and R_Z in equation (2-45).

Molodensky-Badekas is sometimes quoted in a partially-linear form in which only the rotation matrix has been linearised. The abbreviation used here is PLMB, and the transformation formula is

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} X_m \\ Y_m \\ Z_m \end{bmatrix} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + (1 + \Delta S) \begin{bmatrix} 1 & -R_Z & R_Y \\ R_Z & 1 & -R_X \\ -R_Y & R_X & 1 \end{bmatrix} \begin{bmatrix} X_s - X_m \\ Y_s - Y_m \\ Z_s - Z_m \end{bmatrix}. \tag{2-47}$$

Deakin (2006), Iliffe and Lott (2008) and Kutoglu (2009a) are among the sources which prefer this form. As with Bursa-Wolf, the partially-linear form would be linearised further when the parameters are optimised by least-squares: the products $\Delta S R_X$, $\Delta S R_Y$ and $\Delta S R_Z$ are deemed to be negligible, which they usually are. In effect, the fully-linear form is used to derive the optimum parameters even when the partially-linear form to apply the transformation.

If the rotations in (2-44) and (2-47) are given the superscripts FLMB and PLMB respectively, then a necessary and sufficient condition for the formulae to be equivalent is that

$$\begin{bmatrix} R_X^{\mathrm{FLMB}} \\ R_Y^{\mathrm{FLMB}} \\ R_Z^{\mathrm{FLMB}} \end{bmatrix} = (1 + \Delta S) \begin{bmatrix} R_X^{\mathrm{PLMB}} \\ R_Y^{\mathrm{PLMB}} \\ R_Z^{\mathrm{PLMB}} \end{bmatrix}. \tag{2-48}$$

This means that the FLMB rotations are less accurate approximations to the true rotations than the PLBW rotations, in the sense that the scaling introduces an additional source of error.

Another consequence of the relationship between the two forms of Molodensky-Badekas is that (2-48) can be used as a substitution to convert PLMB into a fully-linear form.

2.10 8-parameter affine transformation

This transformation was proposed in Andrei (2006). This transformation is based on 3 translation components, 3 rotations and 2 scale factors. The scale factors are for "vertical" distance and distance in the "horizontal" plane. The rationale behind this is that older geodetic datums used different measuring techniques for horizontal networks and levelling networks. Horizontally, in the plane OX′Y′, the transformation is near-conformal.

This requires a change in each coordinate system to a "local level system" discussed in subsection 1.4.4 above. However, equation (1-24) uses a local point (ϕ_0, λ_0, h_0) and its geocentric Cartesian coordinates (X_0, Y_0, Z_0). Given that there are two datums involved, it is necessary to have local points $(\phi_{s,0}, \lambda_{s,0}, h_{s,0})$ & $(\phi_{t,0}, \lambda_{t,0}, h_{t,0})$, in the respective datums and their corresponding geocentric Cartesian coordinates $(X_{s,0}, Y_{s,0}, Z_{s,0})$ & $(X_{t,0}, Y_{t,0}, Z_{t,0})$. The

conversion method used by Andrei (2006) is to compute the Cartesian coordinates of the local points from the average values from the coordinates in the dataset; $(\phi_{s,0}, \lambda_{s,0}, h_{s,0})$ and $(\phi_{t,0}, \lambda_{t,0}, h_{t,0})$ are then computed by Cartesian-to-geodetic conversion (*eg* by the method described in subsection 1.4.1).

The adaptation of (1-24) for converting (X, Y, Z) to local level coordinates (X', Y', Z') in the source and target datums is as follows.

$$\begin{bmatrix} X'_s \\ Y'_s \\ Z'_s \end{bmatrix} = \begin{bmatrix} -\sin\lambda_{s,0} & \cos\lambda_{s,0} & 0 \\ -\sin\phi_{s,0}\cos\lambda_{s,0} & -\sin\phi_{s,0}\sin\lambda_{s,0} & \cos\phi_{s,0} \\ \cos\phi_{s,0}\cos\lambda_{s,0} & \cos\phi_{s,0}\sin\lambda_{s,0} & \sin\phi_{s,0} \end{bmatrix} \begin{bmatrix} X_s - X_{s,0} \\ Y_s - Y_{s,0} \\ Z_s - Z_{s,0} \end{bmatrix} \quad ; \tag{2-49}$$

$$\begin{bmatrix} X'_t \\ Y'_t \\ Z'_t \end{bmatrix} = \begin{bmatrix} -\sin\lambda_{t,0} & \cos\lambda_{t,0} & 0 \\ -\sin\phi_{t,0}\cos\lambda_{t,0} & -\sin\phi_{t,0}\sin\lambda_{t,0} & \cos\phi_{t,0} \\ \cos\phi_{t,0}\cos\lambda_{t,0} & \cos\phi_{t,0}\sin\lambda_{t,0} & \sin\phi_{t,0} \end{bmatrix} \begin{bmatrix} X_t - X_{t,0} \\ Y_t - Y_{t,0} \\ Z_t - Z_{t,0} \end{bmatrix} \quad . \tag{2-50}$$

The reverse coordinate conversions are adaptations of equation (1-25):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \begin{bmatrix} X_{s,0} \\ Y_{s,0} \\ Z_{s,0} \end{bmatrix} + \begin{bmatrix} -\sin\lambda_{s,0} & -\sin\phi_{s,0}\cos\lambda_{s,0} & \cos\phi_{s,0}\cos\lambda_{s,0} \\ \cos\lambda_{s,0} & -\sin\phi_{s,0}\sin\lambda_{s,0} & \cos\phi_{s,0}\sin\lambda_{s,0} \\ 0 & \cos\phi_{s,0} & \sin\phi_{s,0} \end{bmatrix} \begin{bmatrix} X'_s \\ Y'_s \\ Z'_s \end{bmatrix} \quad ; \tag{2-51}$$

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} X_{t,0} \\ Y_{t,0} \\ Z_{t,0} \end{bmatrix} + \begin{bmatrix} -\sin\lambda_{t,0} & -\sin\phi_{t,0}\cos\lambda_{t,0} & \cos\phi_{t,0}\cos\lambda_{t,0} \\ \cos\lambda_{t,0} & -\sin\phi_{t,0}\sin\lambda_{t,0} & \cos\phi_{t,0}\sin\lambda_{t,0} \\ 0 & \cos\phi_{t,0} & \sin\phi_{t,0} \end{bmatrix} \begin{bmatrix} X'_t \\ Y'_t \\ Z'_t \end{bmatrix} \quad . \tag{2-52}$$

The transformation is applied to local level coordinates (X', Y', Z') which are sometimes called ENU coordinates.

$$\begin{bmatrix} X'_t \\ Y'_t \\ Z'_t \end{bmatrix} = \begin{bmatrix} \Delta X' \\ \Delta Y' \\ \Delta Z' \end{bmatrix} + \mathbf{R} \begin{bmatrix} S_h & 0 & 0 \\ 0 & S_h & 0 \\ 0 & 0 & S_v \end{bmatrix} \begin{bmatrix} X'_s \\ Y'_s \\ Z'_s \end{bmatrix} \tag{2-53}$$

The rotation matrix will be in terms of $R_{X'}$, $R_{Y'}$ and $R_{Z'}$ rather than R_X, R_Y and R_Z. Apart from that, it will take the form (2-15) or (2-17), depending on the order of rotations.

The 8-parameter affine transformation examined by Varga *et al* (2017) is, however, in terms of geocentric Cartesian coordinates, making it a special case of the 9-parameter affine transformation. Its scale factors are S_X, S_Y and S_Z, but with $S_Y = S_Z$. There is no obvious rationale for equal scale factors in the plane parallel to the Equatorial plane and a different scale factor in the polar direction. This version of Andrei's transformation will not be considered further, except as an intermediate model in the derivation of 9-parameter affine transformations.

2.11 9-parameter affine transformation

This transformation is based on 3 translation components, 3 rotations and 3 scale factors. The scale factors, S_X, S_Y and S_Z, are in the directions of the axes OX, OY and OZ. The transformation is applied to geocentric Cartesian coordinates (*X, Y, Z*). It is not conformal.

Differences in scale might arise if there are differences between horizontal scale and vertical scale. If, for example, the direction of one of the axes is within 25° of the local vertical directions, then that will be the axis upon which the vertical scale factor has the greatest influence and the horizontal scale factor the least influence.

There are several ways in which the 3 rotations can be combined into rotation matrix **R**. They can be applied rigorously as in equation (2-15) or equation (2-17). Alternatively, the rotation matrix may be in a linearised form as in equation (2-42).

Irrespective of how **R** is defined, the formula for the transformation takes one of two forms.

The "SR" version, which is used by Han (2010), Paláncz *et al* (2008), Späth (2004) and Varga *et al* (2017), is

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} S_X & 0 & 0 \\ 0 & S_Y & 0 \\ 0 & 0 & S_Z \end{bmatrix} \boldsymbol{R} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix}. \tag{2-54}$$

The "RS" version, which is used by Andrei (2006), Fan (2009) and Paláncz and Piroska (2011), is

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{R} \begin{bmatrix} S_X & 0 & 0 \\ 0 & S_Y & 0 \\ 0 & 0 & S_Z \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix}. \tag{2-55}$$

This can alternatively be written as

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{R} \begin{bmatrix} S_X X_s \\ S_Y Y_s \\ S_Z Z_s \end{bmatrix}. \tag{2-56}$$

2.12 12-parameter affine transformation

This transformation is based on 3 translation components and the 9 elements of a 3-by-3 matrix. It is applied to geocentric Cartesian coordinates (X, Y, Z). It is not conformal.

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} \tag{2-57}$$

Normally, a_1, a_5 and a_9 will be close to 1 and the other a_i will be close to 0. The transformation is linear with respect to the parameters, since (2-57) can be rewritten in the following form.

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X_s & Y_s & Z_s & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & X_s & Y_s & Z_s & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & X_s & Y_s & Z_s \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ a_1 \\ \vdots \\ a_9 \end{bmatrix} \tag{2-58}$$

2.13 Multiple regression equations and similar polynomial formulae

This type of transformation refers to equations that express datum shifts as polynomial functions of latitude and longitude. They are called multiple regression equations (MREs) if they are derived in a particular way, which is described in Section 4.13. Essentially, it means restricting the terms used in each polynomial to those which are statistically significant.

Some authors refer to polynomial functions of latitude and longitude as MREs, irrespective of how they are derived. The 6-term equations quoted in Kutoglu (2009b) and Mitsakaki *et al* (2006) appear to have been based on a decision to restrict the terms to those whose powers add up to 2 or less; the 10-term equations quoted in Ayer *et al* (2010) appear to have been based on a decision to restrict the terms to those whose powers add up to 3 or less. By those criteria, these functions are not MREs. However, MREs can be regarded as a sub-type of polynomial formulae, because that is what they are when they are applied.

Ordnance Survey (2002) published formulae with top power 3 for transforming "Irish Grid" to ETRS89 (more accurately, Ireland 1965 to ETRS89). The equations retain all 16 terms without regard for statistical significance, and are described as polynomial transformations rather than MREs. Their adoption for continued use was noted in Ordnance Survey (2020), and they were included in the software package Grid InQuest II which covers datum transformations in Great Britain and Ireland.

Strictly speaking, multiple regression equations are linear combinations of basis functions that need not be monomials or even polynomials. However, all the basis functions used in datum-shift MREs encountered during this research have been monomials (apart from some of the new types introduced in Section 2.17).

The US Defense Mapping Agency (DMA) derived a series of MREs for transforming a number of local datums to the World Geodetic System 1984 (WGS 84). It should be noted that in 1996 DMA was merged with other bodies into what is now the National Geospatial-Intelligence Agency (NGA). Until 2003 NGA was called the National Imagery and Mapping Agency (NIMA).

The MREs in NGA (2014) and NIMA (2000) were originally published by DMA (1991, Appendix D). These formulae are for transformations to WGS84 from "seven major continental datums, covering continental-size land areas with large distortion". A more complete set of MREs covering 54 datum transformations around the world can be found in Sections 19 and 20 of DMA (1987b).

Multiple regression equations for the datum shifts $\Delta\phi$ and $\Delta\lambda$ take the general form

$$\Delta\phi(") = \sum_{i,j} a_{i,j} U^i V^j; \tag{2-59}$$

$$\Delta\lambda(") = \sum_{i,j} b_{i,j} U^i V^j. \tag{2-60}$$

The summations are finite and usually have fewer than 30 terms. U and V are intermediate coordinates which are linear functions of ϕ_s and λ_s respectively. By tradition, the same scaling constant is used for both of ϕ_s and λ_s. The value of this constant, K, is purely a matter of computational convenience; DMA used fractions of π, notably π/60 (=0.05235988) for the continental datums in Appendix D of DMA (1991). The resulting formulae for U and V are:

$$U = K(\phi_{\text{in deg}} - \phi_{\text{off}}), \tag{2-61}$$

$$V = K(\lambda_{\text{in deg}} - \lambda_{\text{off}}), \tag{2-62}$$

where ϕ_{off} and λ_{off} are offsets based on a point near the centre of the area of application.

The formulae for applying the shifts $\Delta\phi$ and $\Delta\lambda$ to the source-datum coordinates are as follows.

$$\phi_t = \phi_s + \Delta\phi; \tag{2-63}$$

$$\lambda_t = \lambda_s + \Delta\lambda. \tag{2-64}$$

MREs of the form (2-59) and (2-60) were first proposed by Appelbaum (1982). He suggested similar summations for shifts in ellipsoidal heights and shifts in Cartesian coordinates. MREs can also be used to model datum shifts for grid coordinates; see Soycan (2005). In all cases, the intermediate variables used in the polynomials are based on latitude and longitude.

Because of the nature of polynomials, it is very important that their use as datum transformation models is restricted to the areas for which they are designed.

2.14 SMITSWAM

SMITSWAM is a new method, or rather an alternative implementation of the rigorous 3PC transformation. It stands for "Standard Molodensky In Two Stages With Applied Misclosure" and was introduced by Ruffhead (2016). It is applied to geocentric Cartesian coordinates (*X, Y, Z*) but produces the results of the rigorous 3PC transformation. This avoids the need to convert geodetic coordinates to Cartesian coordinates in the source datum and Cartesian coordinates to geodetic coordinates in the target datum.

The first application of Standard Molodensky produces the transformation

$$(\phi_s, \lambda_s, h_s) \rightarrow (\phi_t^{SM}, \lambda_t^{SM}, h_t^{SM}). \tag{2-65}$$

The second application can be regarded as "Simple Same-Formula Inverse" as it is from the target datum to the source datum, and the original ΔX, ΔY, ΔZ, Δa and Δf are replaced by their negatives. It produces the transformation

$$\left(\phi_t^{\mathrm{SM}}, \lambda_t^{\mathrm{SM}}, h_t^{\mathrm{SM}}\right) \rightarrow (\phi_s^{\mathrm{SSFI}}, \lambda_s^{\mathrm{SSFI}}, h_s^{\mathrm{SSFI}}). \tag{2-66}$$

The new approximation to the transformed position is obtained by subtracting half the misclosure from the first approximation:

$$\phi_t^{\mathrm{SMITSWAM}} = \phi_t^{\mathrm{SM}} - \left(\phi_s^{\mathrm{SSFI}} - \phi_s\right)/2. \tag{2-67}$$

$$\lambda_t^{\mathrm{SMITSWAM}} = \lambda_t^{\mathrm{SM}} - \left(\lambda_s^{\mathrm{SSFI}} - \lambda_s\right)/2. \tag{2-68}$$

$$h_t^{\mathrm{SMITSWAM}} = h_t^{\mathrm{SM}} - \left(h_s^{\mathrm{SSFI}} - h_s\right)/2. \tag{2-69}$$

Ruffhead (2016) showed that for four local datums being transformed to WGS84, these approximations agreed closely with the conformal model. The revised computations for this thesis, using the test points in Figure 2-4, show SMITSWAM approximations agree with the conformal model to within maximum differences of 0.0000166m in latitude, 0.0000001m in longitude, 0.0000065m in height and 0.0000179m in 3D distance. The RMS of the 3D distance differences is 0.0000092m.

Section 2.3 considered the possibility that ellipsoidal heights are either dummy values or poor estimates. The equivalence of 3PC and SMITSWAM means that Table 2-1 applies to SMITSWAM. This demonstrates that ϕ_t and λ_t from SMITSWAM are relatively insensitive to uncertainty about h_s.

2.15 Variations on Standard Molodensky transformation

The partially-conformal variation (PCV) on Standard Molodensky is a new method with up to 7 parameters, not counting Δa or Δf. It is applied to geodetic coordinates. The full set of parameters consists of

- translation parameters ΔX_{hor}, ΔY_{hor}, ΔZ_{hor} for use in the horizontal-shift equations (2-7) and (2-9);
- rotation parameter R_Z for application to the longitude after the Standard Molodensky longitude formula;
- translation parameters ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} for use in the vertical-shift equation (2-10).

If all 7 parameters are used, the transformation formulae are

$$\begin{aligned}\Delta\phi = [&-\Delta X_{hor}\sin\phi_s\cos\lambda_s - \Delta Y_{hor}\sin\phi_s\sin\lambda_s\\ &+\Delta Z_{hor}\cos\phi_s + \Delta a(\nu_s e_s^2\sin\phi_s\cos\phi_s)/a_s\\ &+\Delta f(\rho_s a_s/b_s + \nu_s b_s/a_s)\sin\phi_s\cos\phi_s]/(\rho_s + h_s).\end{aligned} \tag{2-70}$$

$$\Delta\lambda = R_Z + \frac{-\Delta X_{hor}\sin\lambda_s + \Delta Y_{hor}\cos\lambda_s}{(\nu_s + h_s)\cos\phi_s}. \tag{2-71}$$

$$\begin{aligned}\Delta h = &\Delta X_{ver}\cos\phi_s\cos\lambda_s + \Delta Y_{ver}\cos\phi_s\sin\lambda_s\\ &+\Delta Z_{ver}\sin\phi_s - \Delta a(a_s/\nu_s) + \Delta f(b_s/a_s)\nu_s\sin^2\phi_s.\end{aligned} \tag{2-72}$$

The rationale behind this method is that ΔX_{hor}, ΔY_{hor}, ΔZ_{hor}, R_Z are chosen to minimise the horizontal residual shifts and that ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} are chosen to minimise the vertical residual shifts. How this is done is described in Section 4.15.

A case study involving 44 points common to OSGB36 and WGS84 showed that the improvement in accuracy relative to Standard Molodensky can be as much as 69%.

The transformation has the following properties:

- Horizontally, it is as conformal as Standard Molodensky, which means it is near-conformal horizontally.
- The equations are linear with respect to the parameters, with no approximation of the Z-rotation.
- The equations bypass Cartesian coordinates.
- The equations involve no more computations than Standard Molodensky apart from the addition of the rotation parameter.

There are two special cases of this generalised transformation:

- The 4-parameter transformation based on ΔX, ΔY, ΔZ and rotation R_Z; the same translation parameters are either used in the horizontal & vertical equations or the vertical equation is not used.
- The 6-parameter transformation based on ΔX_{hor}, ΔY_{hor}, ΔZ_{hor}, ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} with no rotation.

The 7-parameter transformations in Sections 2.6 and 2.8 both included the special case of 4 parameters consisting of 3 translations and a Z-rotation. However, in both cases, the rotation was applied before the translations. In addition, Bursa-Wolf applies the rotation approximately.

2.16 Variations on Abridged Molodensky transformation

The partially-conformal variation (PCV) on Abridged Molodensky is a new method with up to 7 parameters, not counting Δa or Δf. It is applied to geodetic coordinates. The full set of parameters consists of

- translation parameters ΔX_{hor}, ΔY_{hor}, ΔZ_{hor} for use in the horizontal-shift equations (2-11) and (2-12);
- rotation parameter R_Z for application to the longitude after the Abridged Molodensky longitude formula;
- translation parameters ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} for use in the vertical-shift equation (2-13).

If all 7 parameters are used, the transformation formulae are

$$\Delta\phi = [-\Delta X_{hor} \sin\phi_s \cos\lambda_s - \Delta Y_{hor} \sin\phi_s \sin\lambda_s + \Delta Z_{hor} \cos\phi_s + (a_s \Delta f + f_s \Delta a) \sin 2\phi_s]/\rho_s. \quad (2\text{-}73)$$

$$\Delta\lambda = R_Z + \frac{-\Delta X_{hor} \sin\lambda_s + \Delta Y_{hor} \cos\lambda_s}{\nu_s \cos\phi_s}. \quad (2\text{-}74)$$

$$\Delta h = \Delta X_{ver} \cos\phi_s \cos\lambda_s + \Delta Y_{ver} \cos\phi_s \sin\lambda_s + \Delta Z_{ver} \sin\phi_s + (a_s \Delta f + f_s \Delta a) \sin^2\phi_s - \Delta a. \quad (2\text{-}75)$$

The rationale behind this method is that ΔX_{hor}, ΔY_{hor}, ΔZ_{hor}, R_Z are chosen to minimise the horizontal residual shifts and that parameters ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} are chosen to minimise the vertical residual shifts. How this is done is described in Section 4.16.

A case study involving 44 points common to OSGB36 and WGS84 showed that the improvement in accuracy relative to Abridged Molodensky can be as much as 69%.

The transformation has the following properties:

- Horizontally, it is as conformal as Abridged Molodensky, which means it is near-conformal horizontally.
- The equations are linear with respect to the parameters, with no approximation of the Z-rotation.
- The equations bypass Cartesian coordinates.
- The equations involve no more computations than Abridged Molodensky apart from the addition of the rotation parameter.

There are two special cases of this generalised transformation:

- The 4-parameter transformation based on ΔX, ΔY, ΔZ and rotation R_Z; the same translation parameters are either used in the horizontal & vertical equations or the vertical equation is not used.
- The 6-parameter transformation based on ΔX_{hor}, ΔY_{hor}, ΔZ_{hor}, ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} with no rotation.

The 7-parameter transformations in Sections 2.6 and 2.8 both included the special case of 4 parameters consisting of 3 translations and a *Z*-rotation. However, in both cases, the rotation was applied before the translations. In addition, Bursa-Wolf applies the rotation approximately.

2.17 Fully-normalised multiple regression equations (5 types)

Ruffhead (2018) noted the widespread practice of using a single scaling factor to obtain the intermediate coordinates *U* and *V*, even when the change in latitude between the area's extremities in degrees differs from the change in longitude. This means that either or both of *U* and *V* could vary outside the range -1 to 1.

Some MREs have been chosen with an unnecessarily small scaling constant, resulting in coefficients that are unnecessarily large. One example is the HDKS→HTRS96 transformation for Croatia in Varga *et al* (2017). The constant 3 is applied to the values in radians of the relative latitude and relative longitude. The result is that *U* varies from -0.11815 to 0.09129 and *V* varies from -0.15327 to 0.16089.

This study favours full normalisation of multiple regression equations in the sense that the intermediate coordinates *U* and *V* are fully normalised in the region of interest. In general,

this means that – when defining the intermediate coordinates – *two* scaling constants are used. This was recommended for the development of future MREs by Ruffhead (2018).

$$U = K_1(\phi_{\text{in deg}} - \phi_{\text{off}}); \tag{2-76}$$

$$V = K_2(\lambda_{\text{in deg}} - \lambda_{\text{off}}). \tag{2-77}$$

This would permit true normalisation of U and V because the scaling constants could be defined to ensure that *both* intermediate coordinates span the range -1 to 1 in the region of interest, as illustrated in Figure 2-7. That would provide a computational justification for deriving the coefficients in the polynomial formulae to the same number of decimal places, since all the monomials U^iV^j would:

- span the range 0 to 1 when i and j are both even;
- span the range -1 to 1 in all other cases.

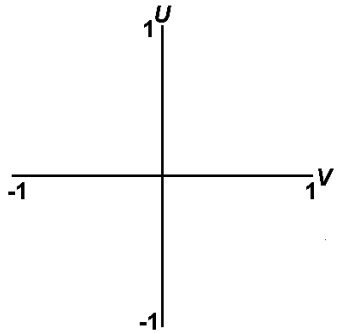


Figure 2-7: Axes for fully-normalised U and V based on the extremities of the region of interest.

In the case of Croatia, noted above, the derivation of scaling factors would be as follows. The latitude limits are approximately 42.5°N and 46.5°N, a latitude range of 2° either side of 44.5°N. The longitude limits are approximately 013.5°E and 019.5°E, a longitude range of 3° either side of 016.5°E. The natural choice of intermediate coordinates would therefore be

$$U = 0.5(\phi_{\text{in deg}} - 44.5) \tag{2-78}$$

$$V = 0.333333(\lambda_{\text{in deg}} - 16.5) \tag{2-79}$$

Fully-normalised multiple regression equations can take the form of linear combinations of monomials U^iV^j as in (2-59) and (2-60). This is, in fact, one of five types described in this section. The other four variations are completely new. Three of them have the characteristic that some of the monomials are specific to a sub-region of the area shown in Figure 2-7. That means they can be used more than once with different coefficients, and all three types can be created without discontinuities in the MREs or their partial derivatives.

The justification for regional variations is they allow more low-order monomials in the MREs and reduce the need for high-order MREs. They represent a compromise between polynomials and spline functions, although the only "joins" occur along either or both of the axes shown in Figure 2-7.

A further type of multiple regression equation uses Chebyshev polynomials as basis functions instead of monomials. These are polynomials bounded by -1 and 1 over the interval [-1,1] but they oscillate between the limits in a similar fashion to trigonometric functions. They are described and illustrated in Appendix F.

In general, an ordinary MRE can be converted into a Chebyshev MRE and vice versa. The Chebyshev MRE with all possible terms up to degree *n* that fits data in a least-squares sense should be equivalent to the ordinary MRE with all possible terms up to degree *n* that fits the same data. However, the formulae will cease to be equivalent when terms deemed statistically insignificant are removed.

The five types described below can be used in combinations. The types used for the latitude shift and the longitude shift do not necessarily have to be the same.

2.17.1 Ordinary multiple regression equations (Ord MREs)

In this instance, the MREs of the normalised *U* and *V* can be expressed as follows:

$$\Delta\phi(") = \sum_{i=0}^{n}\sum_{j=0}^{n} a_{i,j}U^iV^j; \quad (2\text{-}80)$$

$$\Delta\lambda(") = \sum_{i=0}^{n}\sum_{j=0}^{n} b_{i,j}U^iV^j. \quad (2\text{-}81)$$

In each case, there are theoretically $(n+1)^2$ terms, where n is the highest power of *U* or *V*. In practice, several of the coefficients are likely to be zero. The reason for that is either the elimination of statistically insignificant terms or a decision to limit the number of high-order terms. It could be a combination of both reasons.

Table 2-4: Maximum size of an ordinary multiple regression equation

Maximum power of U and V	Maximum number of terms
1	4
2	9
3	16
4	25
5	36
6	49
7	64
8	81
9	100

One consequence of (2-80), (2-81) and the inequality $\left|U^i V^j\right| \leqslant 1$ is that the L_1 norm of coefficients provides an upper bound on the magnitude of the computed shift:

$$|\Delta\phi('')| \leqslant \sum |a_{i,j}|; \tag{2-82}$$

$$|\Delta\lambda('')| \leqslant \sum |b_{i,j}|. \tag{2-83}$$

In reality, these upper bounds should be treated with caution. It will be shown in the case studies that the computed shifts from Ordinary MREs tend to be much smaller than the L_1 norms suggest.

2.17.2 North/south multiple regression equations (N/S MREs)

In this instance, the MREs of the normalised U and V can be expressed as follows:

$$\Delta\phi('') = \sum_{i=0}^{1}\sum_{j=0}^{n} a_{i,j}U^iV^j + \sum_{i=2}^{n}\sum_{j=0}^{n} a_{i,j,N}[U^iV^j]_N + \sum_{i=2}^{n}\sum_{j=0}^{n} a_{i,j,S}[U^iV^j]_S; \tag{2-84}$$

$$\Delta\lambda('') = \sum_{i=0}^{1}\sum_{j=0}^{n} b_{i,j}U^iV^j + \sum_{i=2}^{n}\sum_{j=0}^{n} b_{i,j,N}[U^iV^j]_N + \sum_{i=2}^{n}\sum_{j=0}^{n} b_{i,j,S}[U^iV^j]_S. \tag{2-85}$$

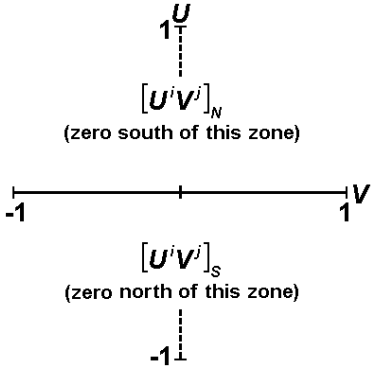


Figure 2-8: Monomials specific to north and south zones, for use in N/S MREs.

The subscripts N and S used on the square-bracketed monomials indicate that they are only non-zero (respectively) north and south of the V-axis. This is illustrated in Figure 2-8. The zone-specific monomials are all divisible by U^2, so their values and partial derivatives with respect to U are zero along the V-axis. This ensures that both functions have C^1 continuity.

In each case, there are theoretically $2n(n+1)$ terms, where n is the highest power of U or V. As with Ord MREs, several of the coefficients are likely to be zero.

Table 2-5: Maximum size of a north/south multiple regression equation

Maximum power of U and V	Maximum number of ordinary monomials	Maximum number of zone-specific monomials	Maximum number of terms
2	6	6	12
3	8	16	24
4	10	30	40
5	12	48	60
6	14	70	84

N/S MRE models can be assumed to be 1.1% faster than Ord MRE models. This is because N/S models are wholly analogous to those discussed in the next subsection for which timed computations were carried out.

2.17.3 East/west multiple regression equations (E/W MREs)

In this instance, the MREs of the normalised U and V can be expressed as follows:

$$\Delta\phi(") = \sum_{i=0}^{n}\sum_{j=0}^{1} a_{i,j}U^iV^j + \sum_{i=0}^{n}\sum_{j=2}^{n} a_{i,j,E}[U^iV^j]_E + \sum_{i=0}^{n}\sum_{j=2}^{n} a_{i,j,W}[U^iV^j]_W; \quad (2\text{-}86)$$

$$\Delta\lambda(") = \sum_{i=0}^{n}\sum_{j=0}^{1} b_{i,j}U^iV^j + \sum_{i=0}^{n}\sum_{j=2}^{n} b_{i,j,E}[U^iV^j]_E + \sum_{i=0}^{n}\sum_{j=2}^{n} b_{i,j,W}[U^iV^j]_W. \quad (2\text{-}87)$$

The subscripts E and W used on the square-bracketed monomials indicate that they are only non-zero (respectively) east and west of the U-axis. This is illustrated in Figure 2-9. The zone-specific monomials are all divisible by V^2, so their values and partial derivatives with respect to V are zero along the U-axis. This ensures that both functions have C^1 continuity.

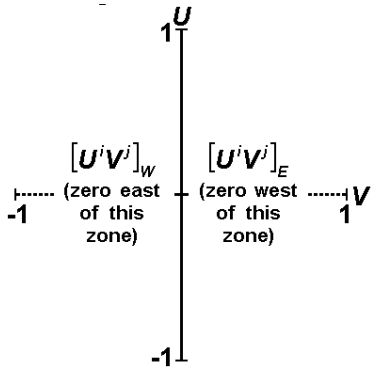


Figure 2-9: Monomials specific to east and west zones, for use in E/W MREs.

In each case, there are theoretically $2n(n+1)$ terms, where n is the highest power of U or V. As with Ord MREs, several of the coefficients are likely to be zero.

Table 2-6: Maximum size of an east/west multiple regression equation

Maximum power of U and V	Maximum number of ordinary monomials	Maximum number of zone-specific monomials	Maximum number of terms
2	6	6	12
3	8	16	24
4	10	30	40
5	12	48	60
6	14	70	84

Timed computations were carried out to compare E/W MREs models with Ord MREs with the same number of terms. The former was 1.1% faster. The E/W MREs' overhead of selecting sector and formulae is more than offset by the fact that they use fewer terms than Ord MREs at any given point.

2.17.4 Four-quadrant multiple regression equations (4Q MREs)

In this instance, the MREs of the normalised U and V can be expressed as follows:

$$\begin{aligned}\Delta\phi(") = {} & \sum_{i=0}^{1}\sum_{j=0}^{n} a_{i,j}U^iV^j + \sum_{i=2}^{n}\sum_{j=0}^{1} a_{i,j}U^iV^j + \\ & \sum_{i=2}^{n}\sum_{j=2}^{n} a_{i,j,NE}[U^iV^j]_{NE} + \sum_{i=2}^{n}\sum_{j=2}^{n} a_{i,j,SE}[U^iV^j]_{SE} + \\ & \sum_{i=2}^{n}\sum_{j=2}^{n} a_{i,j,SW}[U^iV^j]_{SW} + \sum_{i=2}^{n}\sum_{j=2}^{n} a_{i,j,NW}[U^iV^j]_{NW}\ ;\end{aligned} \tag{2-88}$$

$$\begin{aligned}\Delta\lambda(") = {} & \sum_{i=0}^{1}\sum_{j=0}^{n} b_{i,j}U^iV^j + \sum_{i=2}^{n}\sum_{j=0}^{1} b_{i,j}U^iV^j + \\ & \sum_{i=2}^{n}\sum_{j=2}^{n} b_{i,j,NE}[U^iV^j]_{NE} + \sum_{i=2}^{n}\sum_{j=2}^{n} b_{i,j,SE}[U^iV^j]_{SE} + \\ & \sum_{i=2}^{n}\sum_{j=2}^{n} b_{i,j,SW}[U^iV^j]_{SW} + \sum_{i=2}^{n}\sum_{j=2}^{n} b_{i,j,NW}[U^iV^j]_{NW}.\end{aligned} \tag{2-89}$$

The subscripts NE, SE, SE and NW used on the square-bracketed monomials indicate the quadrants in which they are non-zero. This is illustrated in Figure 2-10. The quadrant-specific monomials are all divisible by U^2V^2, so their values and partial derivatives are zero along each axis. This ensures that both functions have C^1 continuity.

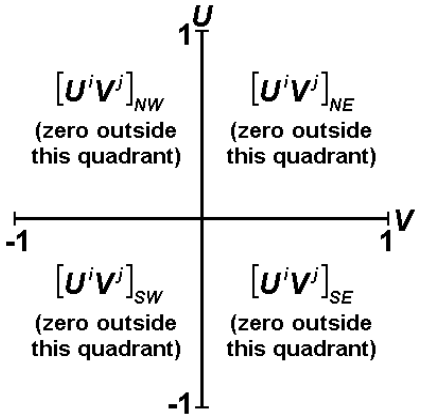


Figure 2-10: Monomials specific to quadrants, for use in 4Q MREs.

In each case, there are theoretically $4(n^2 - n + 1)$ terms, where n is the highest power of U or V. As with Ord MREs, several of the coefficients are likely to be zero.

Table 2-7: Maximum size of a four-quadrant multiple regression equation

Maximum power of U and V	Maximum number of ordinary monomials	Maximum number of quadrant-specific monomials	Maximum number of terms
2	8	4	12
3	12	16	28
4	16	36	52
5	20	64	84

Timed computations were carried out to compare 4Q MREs models with Ord MREs with the same number of terms. The former was 3.5% faster. The 4Q MREs' overhead of selecting quadrant and formulae is more than offset by the fact that they use fewer terms than Ord MREs at any given point.

2.17.5 Chebyshev multiple regression equations (Cheb MREs)

In this instance, the MREs of the normalised U and V can be expressed as follows:

$$\Delta\phi(") = \sum_{i=0}^{n}\sum_{j=0}^{n} a_{i,j}T_i(U)T_j(V); \quad (2\text{-}90)$$

$$\Delta\lambda(") = \sum_{i=0}^{n}\sum_{j=0}^{n} b_{i,j}T_i(U)T_j(V). \quad (2\text{-}91)$$

The terms $T_i(x)$ are Chebyshev polynomials of degree i, and are defined in Appendix F. In each of the above equations, there are theoretically $(n+1)^2$ terms, where n is the highest power of U or V (and also the highest degree of the Chebyshev polynomials). In practice, several of the coefficients are likely to be zero. That could be the result of terms being eliminated as statistically insignificant, or it could be the result of a decision to limit the number of high-order terms. It could be a combination of both reasons.

Table 2-8: Maximum size of a Chebyshev multiple regression equation

Maximum power of U and V	Maximum number of terms
1	4
2	9
3	16
4	25
5	36
6	49
7	64
8	81
9	100

Computational tests on Cheb MREs showed that they take up about 2.5 times the processing times of Ord MREs with the same number of terms. This is because generating Chebyshev polynomials involves more arithmetic than generating monomials.

One consequence of (2-90), (2-91) and the inequality $\left|T_i(U)T_j(U)\right| \leqslant 1$ is that the L_1 norm of coefficients provides an upper bound on the magnitude of the computed shift:

$$|\Delta\phi('')| \leqslant \sum |a_{i,j}|; \tag{2-92}$$

$$|\Delta\lambda('')| \leqslant \sum |b_{i,j}|. \tag{2-93}$$

This means that a shift computed from an ordinary MRE will have its magnitude bounded by the L_1 norm of the coefficients of the equivalent Chebyshev MRE. The case studies will show this is generally much lower that the L_1 norm of the coefficients of the ordinary MRE.

CHAPTER 3: REVERSE TRANSFORMATIONS

This chapter considers the reversibility of different methods. A transformation from one geodetic datum to another is usually an approximate model which does not fit points known in both systems exactly. A reverse (or inverse) transformation will – at best – be equally inaccurate. The best that can be achieved is total consistency between the forward and reverse transformations, in the sense that one followed by the other will result in the original coordinates with no misclosure.

Iliffe and Lott (2008) also discusses whether various transformations are reversible. However, their yardstick for reversibility is whether "the same formula can be used for converting from system A to system B and from B to A, but with some or all of the transformation applied with the opposite sign" (page 133). The yardstick used in this chapter is whether there exists a reverse formula which is computationally accurate relative to the forward method.

For the purpose of this chapter, a reverse transformation is exact if it is totally consistent with the forward method, and approximate if it is not. In other words, the accuracy being described is accuracy relative to the forward transformation.

One type of inverse is a same-formula inverse (SFI) where the same formula is applied to compute the inverse but with different parameters. It is not always possible to find parameters that give an exact result. Sometimes a reverse transformation can be obtained – usually with approximate results – by applying the original formula(e) with the signs of the original parameters reversed. This study follows Ruffhead and Whiting (2020) in designating that as a *simple same-formula inverse* (SSFI).

In several cases, the reverse transformation is expressed as a rearrangement-type formula. This uses the very basic principle that if $y = f\big(g(x)\big)$ and f & g are invertible then $x = g^{-1}\big(f^{-1}(y)\big)$. This applies to relationships between scalar quantities, relationships between vectors and relationships between matrices.

In this chapter, the subscripts *s* and *t* are used to indicate the source datum and target datum of the *original* transformation. In the context of *inverse* transformations, coordinates with subscript *t* are mapped into coordinates with subscript *s*.

3.1 Conformal transformation in 2 dimensions

The reverse transformation $(x_t, y_t) \rightarrow (x_s, y_s)$ can be obtained exactly by rearranging equation (2-1) with a reversal of the rotation:

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \frac{1}{S} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_t - \Delta x \\ y_t - \Delta y \end{bmatrix} \tag{3-1}$$

An equivalent form can be found by rearranging equation (2-2) and applying the inverse matrix:

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \frac{1}{a_1^2 + a_2^2} \begin{bmatrix} a_1 & a_2 \\ -a_2 & a_1 \end{bmatrix} \begin{bmatrix} x_t - \Delta x \\ y_t - \Delta y \end{bmatrix} \tag{3-2}$$

If the coordinate-frame rotation convention of Figure 2-2 is used instead of the position-vector convention of Figure 2-1, then each of the square matrices in equations (3-1) and (3-2)) is replaced by its transpose.

3.2 Affine transformation in 2 dimensions

The reverse transformation $(x_t, y_t) \rightarrow (x_s, y_s)$ can be obtained exactly by rearranging equation (2-4) and applying the inverse matrix:

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \frac{1}{a_1 a_4 - a_2 a_3} \begin{bmatrix} a_4 & -a_2 \\ -a_3 & a_1 \end{bmatrix} \begin{bmatrix} x_t - \Delta x \\ y_t - \Delta y \end{bmatrix}. \tag{3-3}$$

3.3 Rigorous 3-parameter conformal transformation

The inverse transformation simply requires *ΔX*, *ΔY* and *ΔZ* to be of opposite sign to their values for the forward transformation. Alternatively, the original parameters can be subtracted:

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} - \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} \tag{3-4}$$

3.4 Standard Molodensky transformation

Because this is an approximate version of the conformal 3-parameter transformation, negating the values of *ΔX*, *ΔY*, *ΔZ*, *Δa* and *Δf* will not perform the reverse transformation exactly. The reason is that the forward shifts in (2-7), (2-9) and (2-10) depend on the quantities ϕ_s, λ_s and h_s which are not known when the start coordinates are in the "target" datum. The test computations for this thesis established that if the forward transformation is applied to the results thus obtained, the misclosures can be as large as -54 mm in latitude, -93 mm in longitude -32 mm in height, and 100 mm in 3D. However, there is a process (with some similarities to SMITSWAM) that is exact for all practical purposes. Its provisional name is "Inverse Standard Molodensky Via Applied Misclosure" (ISMVAM).

Firstly, the SSFI Standard Molodensky transformation is applied, which means that the original ΔX, ΔY, ΔZ, Δa and Δf are replaced by their negatives. The quantities a, b, ν and ρ will, of course, be computed in terms of the target datum rather than the source datum. (If the ellipsoidal heights are unknown, dummy height values, *eg* zeroes, should be used for this stage.) In terms of notation, the process is described in (3-5).

$$(\phi_t, \lambda_t, h_t) \rightarrow (\phi_s^{\mathrm{SSFI}}, \lambda_s^{\mathrm{SSFI}}, h_s^{\mathrm{SSFI}}). \tag{3-5}$$

Standard Molodensky is then applied to the result.

$$(\phi_s^{\mathrm{SSFI}}, \lambda_s^{\mathrm{SSFI}}, h_s^{\mathrm{SSFI}}) \rightarrow (\phi_t^{\mathrm{SM}}, \lambda_t^{\mathrm{SM}}, h_t^{\mathrm{SM}}). \tag{3-6}$$

The new approximation to the inversely-transformed position is obtained by subtracting the misclosure from the first approximation:

$$\phi_s^{\mathrm{ISMVAM}} = \phi_s^{\mathrm{SSFI}} - (\phi_t^{\mathrm{SM}} - \phi_t). \tag{3-7}$$

$$\lambda_s^{\mathrm{ISMVAM}} = \lambda_s^{\mathrm{SSFI}} - (\lambda_t^{\mathrm{SM}} - \lambda_t). \tag{3-8}$$

$$h_s^{\mathrm{ISMVAM}} = h_s^{\mathrm{SSFI}} - (h_t^{\mathrm{SM}} - h_t). \tag{3-9}$$

Further computation is unnecessary. This conclusion is based on the computations for this thesis using the sets of points in Figure 2-4; see Table 3-1. Applying Standard Molodensky to the corrected approximation produces WGS84 coordinates less than 0.000011m from the originals. The 3D distances between coordinates are less than 0.000012m. ISMVAM should therefore be seen as a corrective process rather than an iterative one. One interesting property is that ISMVAM produces the same errors as the SSFI Standard Molodensky (relative to the conformal 3P transformation) but with opposite sign.

An alternative description of the process is as follows:

- Apply SSFI Standard Molodensky to the "target" coordinates to obtain the first estimate of the "source" coordinates.
- Apply forward Standard Molodensky to the first estimate to check its accuracy.
- Set the misclosure to the "check" coordinates minus the original coordinates.
- Subtract the misclosure from the first estimate to obtain a corrected estimate.

Table 3-1: Accuracy of Inverse Standard Molodensky Process Via Applied Misclosure (measured by misclosure after forward Standard Molodensky is applied to corrected estimate)

	Largest 3D misclosure	Largest latitude misclosure	Largest longitude misclosure	Largest height misclosure
OSGB36 ← WGS84	0.0025mm	-0.0014mm	-0.0023mm	-0.0004mm
AGD84 ← WGS84	0.0002mm	-0.0002mm	-0.0002mm	-0.0000mm
Arc 1950 (Zaire) ← WGS84	0.0041mm	-0.0035mm	-0.0001mm	-0.0025mm
Indian Datum ← WGS84	0.0118mm	-0.0049mm	-0.0105mm	-0.0053mm

3.5 Abridged Molodensky transformation

Because this is an approximate version of the conformal 3-parameter transformation, negating the values of ΔX, ΔY, ΔZ, Δa and Δf will not perform the reverse transformation exactly. The reason is that the forward shifts in (2-11), (2--12) and (2-13) depend on the quantities ϕ_s, λ_s and h_s which are not known when the start coordinates are in the "target" datum. The test computations for this thesis established that if the forward transformation is applied to the results thus obtained, the misclosures can be as large as -56 mm in latitude, -96 mm in longitude -33 mm in height, and 104 mm in 3D. However, there is a process (with some similarities to SMITSWAM) that is exact for all practical purposes. Its provisional name is "Inverse Abridged Molodensky Via Applied Misclosure" (IAMVAM).

Firstly, the SSFI Abridged Molodensky transformation is applied, which means that the original ΔX, ΔY, ΔZ, Δa and Δf are replaced by their negatives. The quantities a, b, ν and ρ will, of course, be computed in terms of the target datum rather than the source datum. (If the ellipsoidal heights are unknown, dummy height values, *eg* zeroes, should be used for this stage.) In terms of notation, the process is described in (3-10).

$$(\phi_t, \lambda_t, h_t) \rightarrow (\phi_s^{\text{SSFI}}, \lambda_s^{\text{SSFI}}, h_s^{\text{SSFI}}). \tag{3-10}$$

Applied Molodensky is then applied to the result.

$$(\phi_s^{\text{SSFI}}, \lambda_s^{\text{SSFI}}, h_s^{\text{SSFI}}) \rightarrow (\phi_t^{\text{AM}}, \lambda_t^{\text{AM}}, h_t^{\text{AM}}). \tag{3-11}$$

The new approximation to the inversely-transformed position is obtained by subtracting the misclosure from the first approximation:

$$\phi_s^{\text{IAMVAM}} = \phi_s^{\text{SSFI}} - (\phi_t^{\text{AM}} - \phi_t). \tag{3-12}$$

$$\lambda_s^{\text{IAMVAM}} = \lambda_s^{\text{SSFI}} - (\lambda_t^{\text{AM}} - \lambda_t). \tag{3-13}$$

$$h_s^{\text{IAMVAM}} = h_s^{\text{SSFI}} - (h_t^{\text{AM}} - h_t). \tag{3-14}$$

Further computation is unnecessary. This conclusion is based on the computations for this thesis using the sets of points in Figure 2-4; see Table 3-2. Applying Abridged Molodensky to the corrected approximation produces WGS84 coordinates less than 0.000013m from the originals. The 3D distances between coordinates are less than 0.000014m. IAMVAM should therefore be seen as a corrective process rather than an iterative one.

An alternative description of the process is as follows:

- Apply SSFI Abridged Molodensky to the "target" coordinates to obtain the first estimate of the "source" coordinates.
- Apply forward Abridged Molodensky to the first estimate to check its accuracy.
- Set the misclosure to the "check" coordinates minus the original coordinates.

- Subtract the misclosure from the first estimate to obtain a corrected estimate.

Table 3-2: Accuracy of Inverse Abridged Molodensky Process Via Applied Misclosure (measured by misclosure after forward Abridged Molodensky is applied to corrected estimate)

	Largest 3D misclosure	Largest latitude misclosure	Largest longitude misclosure	Largest height misclosure
OSGB36 ← WGS84	0.0026mm	-0.0014mm	-0.0023mm	-0.0004mm
AGD84 ← WGS84	0.0001mm	0.0000mm	-0.0001mm	-0.0000mm
Arc 1950 (Zaire) ← WGS84	0.0054mm	-0.0046mm	0.0001mm	-0.0028mm
Indian Datum ← WGS84	0.0140mm	-0.0052mm	-0.0128mm	-0.0054mm

3.6 Rigorous 7-parameter conformal transformation

For the Helmert transformation, only the most general case of 7 parameters needs to be considered. The special cases noted in Section 2.6 involve 4, 5 or 6 parameters, which is equivalent to saying that up to 3 of the usual 7 parameters are zero. Such a possibility does not affect the reverse transformation process.

The reverse transformation $(X_t, Y_t, Z_t) \rightarrow (X_s, Y_s, Z_s)$ can be obtained exactly by rearranging (2-14):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \frac{1}{(1+\Delta S)} \mathbf{R}^{-1} \begin{bmatrix} X_t - \Delta X \\ Y_t - \Delta Y \\ Z_t - \Delta Z \end{bmatrix} \tag{3-15}$$

where **R** is given by either (2-16) or (2-18).

Aktuğ (2009, page 48) acknowledges that a matrix formula equivalent to equation (3-15) can be used to compute the reverse transformation. He uses it to derive SFI parameters, that is, parameters that can be used in the basic transformation formula to achieve the inverse. However, it is simpler – and therefore preferable – to turn (3-15) into an implementable form.

The inverse of **R** is its transpose. If **R** is given by (2-15) and (2-16), then its inverse is given by

$$\mathbf{R}_{ZYX}^{-1} = \begin{bmatrix} c_Y c_Z & c_Y s_Z & -s_Y \\ s_X s_Y c_Z - c_X s_Z & c_X c_Z + s_X s_Y s_Z & s_X c_Y \\ s_X s_Z + c_X s_Y c_Z & c_X s_Y s_Z - s_X c_Z & c_X c_Y \end{bmatrix}. \tag{3-16}$$

If **R** is given by (2-17) and (2-18), then its inverse is given by

$$\mathbf{R}_{XYZ}^{-1} = \begin{bmatrix} c_Y c_Z & s_X s_Y c_Z + c_X s_Z & s_X s_Z - c_X s_Y c_Z \\ -c_Y s_Z & c_X c_Z - s_X s_Y s_Z & c_X s_Y s_Z + s_X c_Z \\ s_Y & -s_X c_Y & c_X c_Y \end{bmatrix}. \tag{3-17}$$

Reit (1998, page 404) has a formula similar to equation (3-15), but only applies it to the case where $\mathbf{R} = \mathbf{R}_{ZYX}$.

If the coordinate-frame rotation convention of Figure 2-6 is used instead of the position-vector convention of Figure 2-5, then each sine term in equations (3-16) and (3-17) needs a change of sign.

Grgić *et al* (2016) considered the possibility of using the Helmert transformation in reverse. They noted, quite correctly, that the parameters would be slightly different in magnitude from the forward parameters, so that merely changing the signs would not be enough. Their method of deriving the reverse parameters requires a second least-squares optimisation, and their results are not computationally exact relative to the forward transformation.

Although equation (3-15) provides a simple and exact method of computing the reverse transformation, it is possible to obtain reverse parameters that enable the Helmert formula to be applied in the other direction (target to source coordinates). In other words, given forward parameters ΔX, ΔY, ΔZ, S, R_X, R_Y and R_Z, there are corresponding same-formula inverse parameters ΔX^{SFI}, ΔY^{SFI}, ΔZ^{SFI}, S^{SFI}, R_X^{SFI}, R_Y^{SFI} and R_Z^{SFI} which make the Helmert formula exact relative to the forward transformation. The proof, which has already appeared in Ruffhead (2021b), is given in subsections 3.6.1 and 3.6.2 below.

3.6.1 Same-formula inverse parameters for Version 1 of Helmert

From (3-15) and the fact that **R** is orthogonal,

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \frac{1}{(1+\Delta S)}\,\mathbf{R}^{\mathrm{T}} \begin{bmatrix} -\Delta X \\ -\Delta Y \\ -\Delta Z \end{bmatrix} + \frac{1}{(1+\Delta S)}\,\mathbf{R}^{\mathrm{T}} \begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} \tag{3-18}$$

The algorithm in Section 2.6.1 is used to compute the equivalent Version-2 rotations R'_X, R'_Y and R'_Z, and the corresponding trigonometric ratios c'_X, s'_X, etc.

$$\begin{aligned} \mathbf{R}^{\mathrm{T}} &= \left\{ \begin{bmatrix} c_Z & -s_Z & 0 \\ s_Z & c_Z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_Y & 0 & s_Y \\ 0 & 1 & 0 \\ -s_Y & 0 & c_Y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_X & -s_X \\ 0 & s_X & c_X \end{bmatrix} \right\}^{\mathrm{T}} \\ &= \left\{ \begin{bmatrix} 1 & 0 & 0 \\ 0 & c'_X & -s'_X \\ 0 & s'_X & c'_X \end{bmatrix} \begin{bmatrix} c'_Y & 0 & s'_Y \\ 0 & 1 & 0 \\ -s'_Y & 0 & c'_Y \end{bmatrix} \begin{bmatrix} c'_Z & -s'_Z & 0 \\ s'_Z & c'_Z & 0 \\ 0 & 0 & 1 \end{bmatrix} \right\}^{\mathrm{T}} \\ &= \begin{bmatrix} c'_Z & s'_Z & 0 \\ -s'_Z & c'_Z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c'_Y & 0 & -s'_Y \\ 0 & 1 & 0 \\ s'_Y & 0 & c'_Y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c'_X & s'_X \\ 0 & -s'_X & c'_X \end{bmatrix}. \end{aligned}$$

Since $s'_X = -\sin(-R'_X)$ and $-s'_X = \sin(-R'_X)$, etc, it follows that

$$R_X^{\mathrm{SFI}} = -R'_X, \quad R_Y^{\mathrm{SFI}} = -R'_Y, \quad R_Z^{\mathrm{SFI}} = -R'_Z. \tag{3-19}$$

From (3-18),

$$S^{\text{SFI}} = \frac{1}{1+\Delta S} = 1 + \frac{(-\Delta S)}{1+\Delta S} \tag{3-20}$$

and

$$\begin{bmatrix}\Delta X^{\text{SFI}} \\ \Delta Y^{\text{SFI}} \\ \Delta Z^{\text{SFI}}\end{bmatrix} = \frac{1}{(1+\Delta S)}\,\mathbf{R}^{\text{T}}\begin{bmatrix}-\Delta X \\ -\Delta Y \\ -\Delta Z\end{bmatrix} \tag{3-21}$$

which can be expressed as

$$\begin{bmatrix}\Delta X^{\text{SFI}} \\ \Delta Y^{\text{SFI}} \\ \Delta Z^{\text{SFI}}\end{bmatrix} = \frac{1}{(1+\Delta S)}\begin{bmatrix} c_Y c_Z & c_Y s_Z & -s_Y \\ s_X s_Y c_Z - c_X s_Z & c_X c_Z + s_X s_Y s_Z & s_X c_Y \\ s_X s_Z + c_X s_Y c_Z & c_X s_Y s_Z - s_X c_Z & c_X c_Y \end{bmatrix}\begin{bmatrix}-\Delta X \\ -\Delta Y \\ -\Delta Z\end{bmatrix}. \tag{3-22}$$

This completes the computation of the SFI parameters for Version 1.

Aktuğ (2009, page 48) describes a derivation of the inverse datum transformation, but claims that "the scale and the rotation parameters will be the same as the direct transformation parameters with opposite signs". He is mistaken on both counts. Equation (3-20) disproves his conclusion about scale change, and his consideration of rotations makes no allowance for the order in which they are applied.

If the software for 3D conformal transformations in position-fixing devices is based solely on the basic formula Helmert Version 1, the SFI parameters can be fed into the stored data for reverse transformations. In particular, parameters for WGS84 to local datums can be derived once-and-for-all from the parameters for local datums to WGS84, and then used in the same way.

The method of deriving same-inverse parameters for Version 1 of Helmert can be extended to applications in geomatics where rotations are not necessarily numerically smaller than 90º. The starting point is *any* set of Version-2 rotations R'_X, R'_Y and R'_Z which generate the same rotation matrix (without regard for whether or not they are unique). Table 2-2 covers the exceptional cases for which the main algorithm in subsection 2.6.1 doesn't work. Formulae (3-19) to (3-22) retain their validity in all such cases.

3.6.2 Same-formula inverse parameters for Version 2 of Helmert

Equation (3-15) applies in this case also.

The algorithm in Section 2.6.2 is used to compute the equivalent Version-1 rotations R'_X, R'_Y and R'_Z, and the corresponding trigonometric ratios c'_X, s'_X, etc. The SFI parameters can be obtained from equations (3-19), (3-20) and (3-21), the proofs being entirely analogous to those given in Section 3.6.1. The only difference is that the expanded version of (3-21) is

$$\begin{bmatrix}\Delta X^{\mathrm{SFI}}\\ \Delta Y^{\mathrm{SFI}}\\ \Delta Z^{\mathrm{SFI}}\end{bmatrix} = \frac{1}{(1+\Delta S)}\begin{bmatrix} c_Y c_Z & s_X s_Y c_Z + c_X s_Z & s_X s_Z - c_X s_Y c_Z \\ -c_Y s_Z & c_X c_Z - s_X s_Y s_Z & c_X s_Y s_Z + s_X c_Z \\ s_Y & -s_X c_Y & c_X c_Y \end{bmatrix}\begin{bmatrix}-\Delta X\\ -\Delta Y\\ -\Delta Z\end{bmatrix}. \tag{3-23}$$

This completes the computation of the SFI parameters for Helmert Version 2.

If the software for 3D conformal transformations in position-fixing devices is based solely on the basic formula Helmert Version 2, the SFI parameters can be fed into the stored data for reverse transformations. In particular, parameters for WGS84 to local datums can be derived once-and-for-all from the parameters for local datums to WGS84, and then used in the same way.

The method of deriving same-inverse parameters for Version 2 of Helmert can be extended to applications in geomatics where rotations are not necessarily numerically smaller than 90º. The starting point is *any* set of Version-1 rotations R'_X, R'_Y and R'_Z which generate the same rotation matrix (without regard for whether or not they are unique). Table 2-3 covers the exceptional cases for which the main algorithm in subsection 2.6.2 doesn't work. Formulae (3-19) to (3-21) and (3-23) retain their validity in all such cases.

3.7 Rigorous localised 7-parameter conformal transformation

Only the most general case of 7 parameters needs to be considered. The special cases noted in Section 2.7 involve 4, 5 or 6 parameters, which is equivalent to saying that up to 3 of the usual 7 parameters are zero. Such a possibility that does not affect the reverse transformation process.

The reverse transformation $(X_t, Y_t, Z_t) \rightarrow (X_s, Y_s, Z_s)$ can be obtained exactly by rearranging (2-39):

$$\begin{bmatrix}X_s\\ Y_s\\ Z_s\end{bmatrix} = \begin{bmatrix}X_m\\ Y_m\\ Z_m\end{bmatrix} + \frac{1}{(1+\Delta S)}\mathbf{R}^{-1}\begin{bmatrix}X_t - X_m - \Delta X\\ Y_t - Y_m - \Delta Y\\ Z_t - Z_m - \Delta Z\end{bmatrix} \tag{3-24}$$

where **R** is given by either (2-16) or (2-18). In each case, the inverse of **R** is its transpose, given by (3-16) or (3-17).

When the transformation involves local level coordinates, the process of obtaining X'_s, Y'_s and Z'_s from X'_t, Y'_t and Z'_t is wholly analogous to the reverse-transformation process of Helmert. This covers what is described in Rapp (1993) and Leick (1995) as the Veis transformation. (See subsection 1.4.4 and Section 2.7.)

It should be noted that the inverse datum transformation cannot be based on the assumption that "the scale and the rotation parameters will be the same as the direct transformation parameters with opposite signs". This quote from Aktuğ (2009, page 48) was disproved in Section 3.6 in relation to Helmert, and the same reasons apply here.

3.8 Bursa-Wolf

For the fully-linear version of Bursa Wolf, the reverse transformation $(X_t, Y_t, Z_t) \rightarrow (X_s, Y_s, Z_s)$ can be obtained exactly by rearranging (2-40):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \begin{bmatrix} 1+\Delta S & -R_Z & R_Y \\ R_Z & 1+\Delta S & -R_X \\ -R_Y & R_X & 1+\Delta S \end{bmatrix}^{-1} \begin{bmatrix} X_t - \Delta X \\ Y_t - \Delta Y \\ Z_s - \Delta Z \end{bmatrix}. \tag{3-25}$$

Aktuğ (2009, page 48) acknowledges that a matrix formula equivalent to equation (3-25) can be used to compute the reverse transformation. He uses it to derive SFI parameters, that is, parameters that can be used in the basic transformation formula to achieve the inverse. However, it is simpler – and therefore preferable – to apply equation (3-25) itself as the reverse formula. The exact inverse matrix can be computed from Cramer's Rule, using the substitution $S = 1 + \Delta S$.

$$\begin{bmatrix} S & -R_Z & R_Y \\ R_Z & S & -R_X \\ -R_Y & R_X & S \end{bmatrix}^{-1} = \frac{1}{D} \begin{bmatrix} S^2 + R_X^2 & R_X R_Y + R_Z S & R_X R_Z - R_Y S \\ R_X R_Y - R_Z S & S^2 + R_Y^2 & R_Y R_Z + R_X S \\ R_X R_Z + R_Y S & R_Y R_Z - R_X S & S^2 + R_Z^2 \end{bmatrix} \tag{3-26}$$

where the determinant *D* is given by

$$D = S^3 + (R_X^2 + R_Y^2 + R_Z^2)S. \tag{3-27}$$

If the coordinate-frame rotation convention of Figure 2-6 is used instead of the position-vector convention of Figure 2-5, then each rotation in equations (3-25) and (3-26) needs a change of sign.

For the partially-linear version of Bursa-Wolf, the reverse transformation $(X_t, Y_t, Z_t) \rightarrow (X_s, Y_s, Z_s)$ can be obtained exactly by rearranging (2-42):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \frac{1}{1+\Delta S} \begin{bmatrix} 1 & -R_Z & R_Y \\ R_Z & 1 & -R_X \\ -R_Y & R_X & 1 \end{bmatrix}^{-1} \begin{bmatrix} X_t - \Delta X \\ Y_t - \Delta Y \\ Z_s - \Delta Z \end{bmatrix}. \tag{3-28}$$

The exact inverse matrix can be found by Cramer's Rule:

$$\begin{bmatrix} 1 & -R_Z & R_Y \\ R_Z & 1 & -R_X \\ -R_Y & R_X & 1 \end{bmatrix}^{-1} = \frac{1}{D} \begin{bmatrix} 1 + R_X^2 & R_X R_Y + R_Z & R_X R_Z - R_Y \\ R_X R_Y - R_Z & 1 + R_Y^2 & R_Y R_Z + R_X \\ R_X R_Z + R_Y & R_Y R_Z - R_X & 1 + R_Z^2 \end{bmatrix} \tag{3-29}$$

where the determinant *D* is given by

$$D = 1 + R_X^2 + R_Y^2 + R_Z^2. \tag{3-30}$$

3.9 Molodensky-Badekas

The exact reversibility of Molodensky-Badekas is not widely acknowledged. Knippers (2009) actually states that it "is not reversible". Taking Molodensky-Badekas in its linear form, the reverse transformation $(X_t, Y_t, Z_t) \rightarrow (X_s, Y_s, Z_s)$ can be obtained exactly by rearranging (2-44):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \begin{bmatrix} X_m \\ Y_m \\ X_m \end{bmatrix} + \begin{bmatrix} 1+\Delta S & -R_Z & R_Y \\ R_Z & 1+\Delta S & -R_X \\ -R_Y & R_X & 1+\Delta S \end{bmatrix}^{-1} \begin{bmatrix} X_t - X_m - \Delta X \\ Y_t - Y_m - \Delta Y \\ Z_t - X_m - \Delta Z \end{bmatrix} \tag{3-31}$$

Substituting S for $1+\Delta S$, the exact inverse matrix is given by (3-26) using the determinant in (3-27).

If the coordinate-frame rotation convention of Figure 2-6 is used instead of the position-vector convention of Figure 2-5, then each rotation in equation (3-31) needs a change of sign.

For the partially-linear version of Molodensky-Badekas, the reverse transformation $(X_t, Y_t, Z_t) \rightarrow (X_s, Y_s, Z_s)$ can be obtained exactly by rearranging (2-47):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \begin{bmatrix} X_m \\ Y_m \\ X_m \end{bmatrix} + \frac{1}{1+\Delta S} \begin{bmatrix} 1 & -R_Z & R_Y \\ R_Z & 1 & -R_X \\ -R_Y & R_X & 1 \end{bmatrix}^{-1} \begin{bmatrix} X_t - X_m - \Delta X \\ Y_t - Y_m - \Delta Y \\ Z_s - Z_m - \Delta Z \end{bmatrix}. \tag{3-32}$$

The exact inverse matrix is given by (3-29) using the determinant in (3-30).

3.10 8-parameter affine transformation

As noted in Section 2.10, (X', Y', Z') are ENU coordinates in a local level system. The reverse transformation $(X'_t, Y'_t, Z'_t) \rightarrow (X'_s, Y'_s, Z'_s)$ can be obtained exactly by rearranging (2-53):

$$\begin{bmatrix} X'_s \\ Y'_s \\ Z'_s \end{bmatrix} = \begin{bmatrix} 1/S_h & 0 & 0 \\ 0 & 1/S_h & 0 \\ 0 & 0 & 1/S_v \end{bmatrix} \mathbf{R}^{-1} \begin{bmatrix} X'_t - \Delta X' \\ Y'_t - \Delta Y' \\ Z'_t - \Delta Z' \end{bmatrix} \tag{3-33}$$

where **R** is analogous to (2-15) or (2-17). In this case, the inverse of **R** is its transpose, as it is the product of rotation matrices.

3.11 9-parameter affine transformation

The reverse transformation $(X_t, Y_t, Z_t) \rightarrow (X_s, Y_s, Z_s)$ depends on whether the forward transformation is of type "SR" or "RS".

The reverse transformation of the "SR" version can be obtained exactly by rearranging (2-54):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \mathbf{R}^{-1} \begin{bmatrix} (X_t - \Delta X)/S_X \\ (Y_t - \Delta Y)/S_Y \\ (Z_t - \Delta Z)/S_Z \end{bmatrix}. \tag{3-34}$$

The reverse transformation of "RS" version can be obtained by rearranging (2-55):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \begin{bmatrix} 1/S_X & 0 & 0 \\ 0 & 1/S_Y & 0 \\ 0 & 0 & 1/S_Z \end{bmatrix} \mathbf{R}^{-1} \begin{bmatrix} X_t - \Delta X \\ Y_t - \Delta Y \\ Z_t - \Delta Z \end{bmatrix}. \tag{3-35}$$

The inverse of **R** depends on whether **R** is a rigorous or linearised rotation matrix. If it is a rigorous rotation matrix, as in Section 2.7, the inverse is the transpose of **R**. If **R** is the linearised rotation matrix used in equation (2-47), then its inverse is given by (3-29) and (3-30).

3.12 12-parameter affine transformation

The reverse transformation $(X_t, Y_t, Z_t) \rightarrow (X_s, Y_s, Z_s)$ can be obtained exactly by rearranging (2-57):

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix}^{-1} \begin{bmatrix} X_t - \Delta X \\ Y_t - \Delta Y \\ Z_t - \Delta Z \end{bmatrix} \tag{3-36}$$

The exact inverse matrix can be found by Cramer's Rule:

$$\begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix}^{-1} = \frac{1}{D} \begin{bmatrix} a_5a_9 - a_6a_8 & a_3a_8 - a_2a_9 & a_2a_6 - a_3a_5 \\ a_6a_7 - a_4a_9 & a_1a_9 - a_3a_7 & a_3a_4 - a_1a_6 \\ a_4a_8 - a_5a_7 & a_2a_7 - a_1a_8 & a_1a_5 - a_2a_4 \end{bmatrix} \tag{3-37}$$

where the determinant D is given by

$$D = a_1(a_5a_9 - a_6a_8) + a_2(a_6a_7 - a_4a_9) + a_3(a_4a_8 - a_5a_7). \tag{3-38}$$

3.13 Multiple regression equations

The reverse formulae corresponding to (2-63) and (2-64) are

$$\phi_s = \phi_t - \Delta\phi; \tag{3-39}$$

$$\lambda_s = \lambda_t - \Delta\lambda. \tag{3-40}$$

However, neither of these formulae is explicit, since $\Delta\phi$ and $\Delta\lambda$ are functions of U and V, which are linear functions of ϕ and λ in the source datum. Appelbaum (1982) fully realised this when he computed U and V from ϕ and λ on WGS 72 for the reverse transformation to ED 50. His analysis concluded that none of the differences in the coordinate shifts at his 53 control points between the two directions was greater than 0.02 metres.

Ruffhead (2018) examines various alternatives for computing the reverse transformation. The most accurate method is a "predictor-corrector" algorithm.

- U and V are computed from the target-datum coordinates (as if they were source-datum coordinates).
- Provisional shifts $\Delta\phi$ and $\Delta\lambda$ are computed from (2-59) and (2-60).
- Approximate values of ϕ_s and λ_s are computed from (3-39) and (3-40).
- U and V are re-computed from the approximate values of ϕ_s and λ_s.
- The shifts $\Delta\phi$ and $\Delta\lambda$ are re-computed from (2-63) and (2-64).
- (3-39) and (3-40) are applied to give ϕ_s and λ_s.

The available evidence indicates that no further iteration is necessary. This was certainly the case for the inter-continental MREs in NIMA (2000), for which the root-mean-square horizontal distance error was smaller than 0.01 millimetres. That needs to be qualified in one respect. The area of application for MREs transforming Córrego Alegre to WGS84 is not the "South American mainland" stated in NIMA (2000); it should instead be taken as that region for which NIMA has contoured the datum shifts. Ruffhead (2018) illustrates the Brazilian part of that region in a diagram, and makes the point that the north-west part of Brazil must be excluded due to absence of data.

3.14 SMITSWAM

SMITSWAM produces the same result as the rigorous 3PC transformation and the latter works in reverse. It follows that the method works in reverse.

3.15 Variations on Standard Molodensky transformation

For the 7-parameter Standard Molodensky PCV transformation, the process described in Section 3.4 needs to be adapted. This is because of the rotation parameter is applied last in the forward transformation, so it needs to be applied first in the reverse process. In other words, an intermediate longitude needs to be set to $\lambda_t - R_Z$ before being fed into the reverse process. This will ensure that the Z-rotation has absolutely no effect on the misclosure that is applied to obtain the reverse transformation.

Intermediate longitudes λ_t^{U} (where U stands for "unrotated") are computed using

$$\lambda_t^{\mathrm{U}} = \lambda_t - R_Z. \tag{3-41}$$

The "simple inverse" $(\phi_s^{\mathrm{SI}}, \lambda_s^{\mathrm{SI}}, h_s^{\mathrm{SI}})$ is obtained by computing the following shifts and adding them to ϕ_t, λ_t, h_t. ($\phi_s^{\mathrm{SI}} = \phi_t + \Delta\phi^{\mathrm{SI}}$, etc.)

$$\Delta\phi^{\mathrm{SI}} = [\Delta X_{hor}\sin\phi_t\cos\lambda_t^{\mathrm{U}} + \Delta Y_{hor}\sin\phi_t\sin\lambda_t^{\mathrm{U}} - \Delta Z_{hor}\cos\phi_t - \Delta a(N_t e_t^2\sin\phi_t\cos\phi_t)/a_t - \Delta f(\rho_t a_t/b_t + N_t b_t/a_t)\sin\phi_t\cos\phi_t]/(\rho_t + h_t). \quad (3\text{-}42)$$

$$\Delta\lambda^{\mathrm{SI}} = (\Delta X_{hor}\sin\lambda_t^{\mathrm{U}} - \Delta Y_{hor}\cos\lambda_t^{\mathrm{U}})/[(N_t + h_t)\cos\phi_t]. \quad (3\text{-}43)$$

$$\Delta h^{\mathrm{SI}} = -\Delta X_{ver}\cos\phi_t\cos\lambda_t^{\mathrm{U}} - \Delta Y_{ver}\cos\phi_t\sin\lambda_t^{\mathrm{U}} - \Delta Z_{ver}\sin\phi_t + \Delta a(a_t/N_t) - \Delta f(b_t/a_t)N_t\sin^2\phi_t. \quad (3\text{-}44)$$

Standard Molodensky PCV formulae (2-70)-(2-72) are applied to $(\phi_s^{\mathrm{SI}}, \lambda_s^{\mathrm{SI}}, h_s^{\mathrm{SI}})$, with those coordinates used to derive ρ_s and N_s. The result, $\left(\phi_t^{\mathrm{PCV}}, \lambda_t^{\mathrm{PCV}}, h_t^{\mathrm{PCV}}\right)$, will give a small misclosure $\left(\phi_t^{\mathrm{PCV}} - \phi_t, \lambda_t^{\mathrm{PCV}} - \lambda_t, h_t^{\mathrm{PCV}} - h_t\right)$.

Subtraction of the misclosure from the simple inverse will give a corrected inverse which is as good as exact, with any subsequent misclosure at sub-millimetre level. Its provisional name is "Inverse Standard Molodensky Variation Via Applied Misclosure" (ISMVVAM).

$$\phi_s^{\mathrm{ISMVVAM}} = \phi_s^{\mathrm{SI}} - \left(\phi_t^{\mathrm{PCV}} - \phi_t\right). \quad (3\text{-}45)$$

$$\lambda_s^{\mathrm{ISMVVAM}} = \lambda_s^{\mathrm{SI}} - \left(\lambda_t^{\mathrm{PCV}} - \lambda_t\right). \quad (3\text{-}46)$$

$$h_s^{\mathrm{ISMVVAM}} = h_s^{\mathrm{SI}} - \left(h_t^{\mathrm{PCV}} - h_t\right). \quad (3\text{-}47)$$

3.16 Variations on Abridged Molodensky transformation

For the 7-parameter Abridged Molodensky PCV transformation, the process described in Section 3.5 needs to be adapted. The process described in Section 3.5 needs to be adapted for the same reason that the Standard Molodensky inverse process was adapted in the previous section.

Intermediate longitudes λ_t^{U} (where U stands for "unrotated") are computed using

$$\lambda_t^{\mathrm{U}} = \lambda_t - R_Z. \quad (3\text{-}48)$$

The "simple inverse" $(\phi_s^{\mathrm{SI}}, \lambda_s^{\mathrm{SI}}, h_s^{\mathrm{SI}})$ is obtained by computing the following shifts and adding them to ϕ_t, λ_t, h_t. ($\phi_s^{\mathrm{SI}} = \phi_t + \Delta\phi^{\mathrm{SI}}$, etc.)

$$\Delta\phi^{\mathrm{SI}} = [\Delta X_{hor}\sin\phi_t\cos\lambda_t^{\mathrm{U}} + \Delta Y_{hor}\sin\phi_t\sin\lambda_t^{\mathrm{U}} - \Delta Z_{hor}\cos\phi_t - (a_t\Delta f + f_t\Delta a)\sin 2\phi_t]/\rho_s. \quad (3\text{-}49)$$

$$\Delta\lambda^{\mathrm{SI}} = (\Delta X_{hor}\sin\lambda_t^{\mathrm{U}} - \Delta Y_{hor}\cos\lambda_t^{\mathrm{U}})/[(N_t + h_t)\cos\phi_t]. \quad (3\text{-}50)$$

$$\Delta h^{\mathrm{SI}} = -\Delta X_{ver}\cos\phi_t\cos\lambda_t^{\mathrm{U}} - \Delta Y_{ver}\cos\phi_t\sin\lambda_t^{\mathrm{U}} - \Delta Z_{ver}\sin\phi_t - (a_t\Delta f + f_t\Delta a)\sin^2\phi_t + \Delta a. \quad (3\text{-}51)$$

Abridged Molodensky PCV formulae (2-73)-(2-75) are applied to $(\phi_s^{\mathrm{SI}}, \lambda_s^{\mathrm{SI}}, h_s^{\mathrm{SI}})$, with those coordinates used to derive ρ_s and N_s. The result, $\left(\phi_t^{\mathrm{PCV}}, \lambda_t^{\mathrm{PCV}}, h_t^{\mathrm{PCV}}\right)$, will give a small misclosure $\left(\phi_t^{\mathrm{PCV}} - \phi_t, \lambda_t^{\mathrm{PCV}} - \lambda_t, h_t^{\mathrm{PCV}} - h_t\right)$.

Subtraction of the misclosure from the simple inverse will give a corrected inverse which is as good as exact, with any subsequent misclosure at sub-millimetre level. Its provisional name is "Inverse Abridged Molodensky Variation Via Applied Misclosure" (IAMVVAM).

$$\phi_s^{\mathrm{IAMVVAM}} = \phi_s^{\mathrm{SI}} - \left(\phi_t^{\mathrm{PCV}} - \phi_t\right). \tag{3-52}$$

$$\lambda_s^{\mathrm{IAMVVAM}} = \lambda_s^{\mathrm{SI}} - \left(\lambda_t^{\mathrm{PCV}} - \lambda_t\right). \tag{3-53}$$

$$h_s^{\mathrm{IAMVVAM}} = h_s^{\mathrm{SI}} - \left(h_t^{\mathrm{PCV}} - h_t\right). \tag{3-54}$$

3.17 Fully-normalised multiple regression equations (5 types)

The procedure recommended in 3.13 applies equally here.

CHAPTER 4: DERIVATION OF TRANSFORMATIONS

The starting point for deriving a datum transformation is a set of common points, that is, a set of points with coordinates known with respect to both datums. Ideally, three-dimensional coordinates for the common points will be known in both systems. In practice, ellipsoidal heights are often unknown in at least one of the datums. To derive a three-dimensional model in such cases, approximate ellipsoidal heights of uncertain quality need to be introduced. A model obtained in this way will – at best – only be trustworthy for transforming horizontal coordinates.

The problem of uncertain ellipsoidal heights does not occur in those two-dimensional transformations that do not use Cartesian coordinates.

Deriving transformations is primarily about deriving the parameters of a model which ensure a close fit to datum shifts at known common points. An exact fit may be possible in special cases, but it is more usual for there to be too many observations for the parameters to enable the model to fit them. In general, therefore, some measure of the residuals needs to be minimised. The most usual method is least-squares optimisation.

If the number of common points is large, there is a strong case for setting aside a subset of those points as "test points". The "control points" – that is, those points used for the least-squares optimisation – would be the remaining points. The test points provide an independent check on the accuracy of the mathematical model, although it means that the model is derived from less than the full amount of data. Selection of the test points should aim to ensure that on the one hand they are representative, and that on the other the coverage of the area by the control points is not significantly inferior to that of the common points as a whole. If there are 4000 common points, perhaps 300-400 could be spared for use as test points.

Least-squares optimisation is the process of obtaining the parameters of a model that minimise the sum of the squares of the residuals. More generally, it is the sum of the weighted squares of residuals that is minimised, where the weights take account of any differences in the perceived quality of the observations. Weights are sometimes introduced to harmonise the units used in observation equations, although this is not needed if the equations already have harmonised units.

Least-squares optimisation is easiest when applied to a model which is linear with respect to its parameters. The general form for an over-determined system of linear equations is

$$\mathbf{Ax} + \mathbf{v} = \mathbf{b} \tag{4-1}$$

where **x** is the vector of parameters, **b** is the vector of observed quantities the model is designed to fit and **v** is the vector of residuals. **A** is the "design matrix" based on the model, and its *i*th column consists of the coefficients of the *i*th parameter.

The quantity to be minimised can be expressed as $\mathbf{v}^{\mathrm{T}}\mathbf{W}\mathbf{v}$, where **W** is a square symmetric matrix of weights. If the observations are uncorrelated and weighted equally, then **W** is a constant multiple of the identity matrix, and for practical purposes can be set to the identity matrix. It is well-known (see, for example, Cross, 1983) that the vector **x** which minimises $\mathbf{v}^{\mathrm{T}}\mathbf{W}\mathbf{v}$ is the solution of the following normal equations.

$$\mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{A}\mathbf{x} = \mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{b}. \tag{4-2}$$

The algebraic solution of (4-2) is given below, although computing the inverse matrix $(\mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{A})^{-1}$ is not necessarily the best way to obtain a numerical solution.

$$\mathbf{x} = (\mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{A})^{-1}\mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{b}. \tag{4-3}$$

There are various ways of describing the quality of the least-squares solution:

- a norm of the residual vector **v**, such as the root-mean-square (RMS) of the control-point residuals;
- the standard deviation of the control-point residuals given by (4-4) below where *m* and *n* are the dimensions of **A** (the numbers of observations and parameters respectively);

$$\sigma_0 = \sqrt{\mathbf{v}^{\mathrm{T}}\mathbf{W}\mathbf{v}/(m-n)} \tag{4-4}$$

- a measure of the test-point residuals, such as the RMS, although this is only possible if test points are extracted from the set of common points.

The quantity σ_0 in (4-4) is often referred to as the standard error of an observation of unit weight and its square as the unit variance. In one sense it is not unique; a scaling of **W** would affect the value of σ_0 even though it is obvious from (4-3) that it would have no effect on the estimated parameters. The value of σ_0 is unique if **W** is set to the identity matrix, which is consistently the case in this study.

The variance-covariance matrix of the parameters is as follows (see, for example, Bomford, 1980, sub-appendix D.15):

$$\mathbf{Cov}(\mathbf{x},\mathbf{x}) = \sigma_0^2(\mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{A})^{-1} = \frac{\mathbf{v}^{\mathrm{T}}\mathbf{W}\mathbf{v}}{m-n}(\mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{A})^{-1}. \tag{4-5}$$

This matrix is not affected by any scaling of the matrix **W**. This is because there would be the same cancellation effect as (4-3). A prerequisite for (4-4) and (4-5) to be meaningful is that $m > n$, *ie* the design matrix has more rows than columns so that there is at least one degree of freedom.

It follows from (4-5) that

$$\text{Standard error of } j\text{th parameter} = \text{square root of } j\text{th diagonal element of } \frac{\mathbf{v}^{\mathrm{T}}\mathbf{W}\mathbf{v}}{m-n}(\mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{A})^{-1}. \tag{4-6}$$

Where a datum transformation is based on a provisional combination of parameters, any estimated parameters that are small compared with their standard errors will be candidates for exclusion, on the grounds that they are not statistically significant.

If it is decided to compute the standard errors of the parameters, then it would be necessary to compute $(\mathbf{A}^{\mathrm{T}}\mathbf{W}\mathbf{A})^{-1}$. It has previously been noted that parameters themselves can be obtained without that inverse matrix, since there are other ways of solving equation (4-2).

For this study, an Excel VBA subroutine was written to apply least-squares optimisation to linear observation equations. It can be found in sub-appendix G.2.1.

4.1 Conformal transformation in 2 dimensions

With the parameter-substitution suggested in Section 2.1, this transformation becomes linear, as shown by equation (2-3). Applying it to data point *i*, with residuals indicating a less-than-exact fit, the observation equations are as follows.

$$\begin{bmatrix} x_{t,i} \\ y_{t,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_{s,i} & -y_{s,i} \\ 0 & 1 & y_{s,i} & x_{s,i} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} v_{x,i} \\ v_{y,i} \end{bmatrix} \tag{4-7}$$

Given a set of at least 3 points common to both datums, the full set of observations takes the form below.

$$\begin{bmatrix} x_{t,1} \\ y_{t,1} \\ \vdots \\ x_{t,n} \\ y_{t,n} \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_{s,1} & -y_{s,1} \\ 0 & 1 & y_{s,1} & x_{s,1} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & x_{s,n} & -y_{s,n} \\ 0 & 1 & y_{s,n} & x_{s,n} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} v_{x,1} \\ v_{y,1} \\ \vdots \\ v_{x,n} \\ v_{y,n} \end{bmatrix} \tag{4-8}$$

Since this complies with (4-1), the least-squares estimate of the parameters can be obtained by solving (4-2) or applying (4-3). It is valid for arbitrarily large rotation angles and any positive scale factor.

4.2 Affine transformation in 2 dimensions

As equation (2-5) shows, this model is linear. Applying it to data point *i*, with residuals indicating a less-than-exact fit, the observation equations are as follows.

$$\begin{bmatrix} x_{t,i} \\ y_{t,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_{s,i} & y_{s,i} & 0 & 0 \\ 0 & 1 & 0 & 0 & x_{s,i} & y_{s,i} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} + \begin{bmatrix} v_{x,i} \\ v_{y,i} \end{bmatrix} \tag{4-9}$$

Given a set of at least 4 points common to both datums, the full set of observations takes the form below.

$$\begin{bmatrix} x_{t,1} \\ y_{t,1} \\ \vdots \\ x_{t,n} \\ y_{t,n} \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_{s.1} & y_{s,1} & 0 & 0 \\ 0 & 1 & 0 & 0 & x_{s,1} & y_{s,1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & x_{s,n} & y_{s,n} & 0 & 0 \\ 0 & 1 & 0 & 0 & x_{s,n} & y_{s,n} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} + \begin{bmatrix} v_{x,1} \\ v_{y,1} \\ \vdots \\ v_{x,n} \\ v_{y,n} \end{bmatrix} \tag{4-10}$$

This is in the form given in (4-1); therefore, solution of (4-2) or application of (4-3) will produce the least-squares estimate of the parameters.

A 2-dimensional affine transformation can be derived uniquely from 3 common points provided they form a triangle (meaning that they are not collinear). This is discussed further in Section 6.1.

4.3 Rigorous 3-parameter conformal transformation

Applying (2-6) to data point *i*, with residuals indicating a close rather than exact fit, the observation equations for this model are as follows.

$$\begin{bmatrix} X_{t,i} - X_{s,i} \\ Y_{t,i} - Y_{s,i} \\ Z_{t,i} - Z_{s,i} \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} v_{X,i} \\ v_{Y,i} \\ v_{Z,i} \end{bmatrix} \tag{4-11}$$

For *n* points common to both datums, the full set of observation equations take the form below.

$$\begin{bmatrix} X_{t,1} - X_{s,1} \\ Y_{t,1} - Y_{s,1} \\ Z_{t,1} - Z_{s,1} \\ \vdots \\ X_{t,n} - X_{s,n} \\ Y_{t,n} - Y_{s,n} \\ Z_{t,n} - Z_{s,n} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} v_{X,1} \\ v_{Y,1} \\ v_{Z,1} \\ \vdots \\ v_{X,n} \\ v_{Y,n} \\ v_{Z,n} \end{bmatrix} \tag{4-12}$$

Since this complies with (4-1), the least-squares estimate of the parameters can be obtained by solving (4-2) or applying (4-3).

If the weight matrix is the identity matrix, it can easily be shown that

$$\Delta X = \frac{1}{n}\sum_{i=1}^{n}\left(X_{t,i} - X_{s,i}\right); \tag{4-13}$$

$$\Delta Y = \frac{1}{n}\sum_{i=1}^{n}\left(Y_{t,i} - Y_{s,i}\right); \tag{4-14}$$

$$\Delta Z = \frac{1}{n}\sum_{i=1}^{n}\left(Z_{t,i} - Z_{s,i}\right). \tag{4-15}$$

So with equal weighting, the least-squares estimate of the parameters is the arithmetic mean of the datum shifts at the known points. This result is noted, for example, in Deakin *et al* (1994).

4.4 Standard Molodensky transformation

The easiest way of deriving ΔX, ΔY and ΔZ for Standard Molodensky is to use the least-squares process applicable to the 3PC model or equivalently apply (4-13), (4-14) and (4-15). Page 7-14 of DMA (1987a) says "The ΔX, ΔY, ΔZ datum shifts used to date in the Molodensky Datum Transformation Formulas have normally been mean values". Given that the exact conformal property is going to be sacrificed in any case, it would be more logical to optimise the parameters as used in the Molodensky models, in this case the Standard Molodensky model.

It is possible, however, to rearrange equations (2-7) and (2-10) so that only the multiples of ΔX, ΔY and ΔZ appear on the right-hand side, as is the case for (2-9). Applying these equations to data point *i*, with residuals indicating a less-than-exact fit, the observation equations are as follows. It should be noted that $\Delta\phi_i$ and $\Delta\lambda_i$ are in radians, which ensures that all terms in equations (4-16) to (4-18) are in the same linear units. The radius of curvature in the prime vertical, as defined in equation (1-1), is denoted by N instead of ν. This is to avoid confusion with the residuals.

$$\begin{aligned}(\rho_{s,i} + h_{s,i})\Delta\phi_i - (\Delta a/a_s)(N_i e_s^2 \sin\phi_{s,i}\cos\phi_{s,i}) \\ -\Delta f(\rho_{s,i}a_s/b_s + N_{s,i}b_s/a_s)\sin\phi_{s,i}\cos\phi_{s,i} = -\Delta X \sin\phi_{s,i}\cos\lambda_{s,i} \\ -\Delta Y \sin\phi_{s,i}\sin\lambda_{s,i} + \Delta Z\cos\phi_{s,i} + v_{\phi,i}\end{aligned} \tag{4-16}$$

$$(N_{s,i} + h_{s,i})\Delta\lambda_i \cos\phi_{s,i} = -\Delta X \sin\lambda_{s,i} + \Delta Y\cos\lambda_{s,i} + v_{\lambda,i} \tag{4-17}$$

$$\begin{aligned}\Delta h_i + \Delta a(a_s/N_{s,i}) - \Delta f(b_s/a_s)N_{s,i}\sin^2\phi_{s,i} = \Delta X\cos\phi_{s,i}\cos\lambda_{s,i} \\ +\Delta Y\cos\phi_{s,i}\sin\lambda_{s,i} + \Delta Z\sin\phi_{s,i} + v_{h,i}\end{aligned} \tag{4-18}$$

For n data points, this gives rise to a full set of $3n$ observation equations that comply with (4-1). The least-squares estimate of the parameters can be obtained by solving (4-2) or applying (4-3).

This method of computing the optimum values of ΔX, ΔY and ΔZ requires more work than taking an arithmetic mean of Cartesian shifts, but it is a one-off process. It also provides scope for giving a lower weighting to the height observation equations.

4.5 Abridged Molodensky transformation

As with Standard Molodensky, the easiest way of deriving ΔX, ΔY and ΔZ for Abridged Molodensky is to use the least-squares process applicable to the 3PC model. This is equivalent to applying the mean-value formulae (4-13), (4-14) and (4-15). Given that the exact conformal property is going to be sacrificed in any case, it would be more logical to optimise the parameters as used in the Abridged Molodensky model.

Starting from equations (2-11) to (2-13), observation equations are constructed in the same way as in Section 4.4 (with N denoting the radius of curvature in the prime vertical, and $\Delta\phi_i$ & $\Delta\lambda_i$ in radians).

$$\rho_{s,i}\Delta\phi_i - (a_s\Delta f + f_s\Delta a)\sin 2\phi_{s,i} = -\Delta X\sin\phi_{s,i}\cos\lambda_{s.i} - \Delta Y\sin\phi_{s,i}\sin\lambda_{s,i} + \Delta Z\cos\phi_{s,i} + v_{\phi,i} \quad (4\text{-}19)$$

$$N_{s,i}\Delta\lambda_i\cos\phi_{s,i} = -\Delta X\sin\lambda_{s.i} + \Delta Y\cos\lambda_{s,i} + v_{\lambda,i} \quad (4\text{-}20)$$

$$\Delta h_i - (a_s\Delta f + f_s\Delta a)\sin^2\phi_{s,i} + \Delta a = \Delta X\cos\phi_{s,i}\cos\lambda_{s.i} + \Delta Y\cos\phi_{s,i}\sin\lambda_{s,i} + \Delta Z\sin\phi_{s,i} + v_{h,i} \quad (4\text{-}21)$$

For n data points, this gives rise to $3n$ observation equations expressible in the form (4-1). Solution of (4-2) or application of (4-3) will produce the least-squares estimate of the parameters ΔX, ΔY and ΔZ. The whole process requires more work than applying mean-value formulae, but it is a one-off. It also provides scope for giving a lower weighting to the height observation equations.

There is a special case where equation (4-21) can be disregarded altogether. If ellipsoidal heights are unavailable in either or both of the datums, ΔX, ΔY and ΔZ can be obtained purely from observation equations (4-19) and (4-20). It is important to note that parameters obtained in this way should not be used in the Δh formula (2-13).

Equations (4-19) and (4-20) can be considered a generalisation of Molnár and Timár (2005). They disregarded ellipsoidal heights entirely for the purpose of optimising the horizontal transformation. They scaled equation (2-12) by $\cos\phi$, whereas in this study the observations were equalised fully by conversion to linear units. Ayer and Fosu (2008) used a similar approach to Molnár and Timár to derive ΔX, ΔY and ΔZ from Abridged Molodensky equations for datum transformations in Ghana.

4.6 Rigorous 7-parameter conformal transformation

Deriving the best-fit parameters of the rigorous 7PC (Helmert) model by least-squares is not straightforward because (2-14) is non-linear with respect to the 7 parameters.

The simplest approach is to substitute the optimal Bursa-Wolf parameters in the chosen version of the Helmert transformation. The method of obtaining the optimal Bursa-Wolf parameters is described in Section 4.8. This will fit the control data less well than use of Helmert with the optimal Helmert parameters. However, from the tests undertaken in this study, if the optimal Bursa-Wolf rotation parameters are numerically less than 20″, then the quality of fit is affected by less than 0.1%.

For this study, a method has been devised to address the non-linearity, and hence derive the optimal parameters for the chosen version of Helmert. This is given in Section 5.1 which discusses more advanced model-derivations than those covered in this Chapter.

Section 2.6 noted that linearity occurs in one special case of Helmert. The 4-parameter conformal transformation based on ΔX, ΔY, ΔZ and scale change ΔS is linear with respect to those parameters. Given a set of at least 2 points common to both datums, the set of observations for point i takes the form below.

$$\begin{bmatrix} X_{t,i} - X_{s,i} \\ Y_{t,i} - Y_{s,i} \\ Z_{t,i} - Z_{s,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X_{s,i} \\ 0 & 1 & 0 & Y_{s,i} \\ 0 & 0 & 1 & Z_{s,i} \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \Delta S \end{bmatrix} + \begin{bmatrix} v_{X,i} \\ v_{Y,i} \\ v_{Z,i} \end{bmatrix}. \tag{4-22}$$

For n data points, this gives rise to a full set of $3n$ observation equations that comply with (4-1). The least-squares estimate of the parameters can be obtained by solving (4-2) or applying (4-3).

4.7 Rigorous localised 7-parameter conformal transformation

Deriving the best-fit parameters of the rigorous localised 7-parameter conformal transformation by least-squares is complicated by the fact that (2-39) is non-linear with respect to the 7 parameters. The available options are similar to those for Helmert, except that any part of the process that used Bursa-Wolf linearisation would use Molodensky-Badekas linearisation instead.

Linearity occurs in one special case noted in Section 2.7. The 4-parameter conformal transformation based on ΔX, ΔY, ΔZ and scale change ΔS is linear with respect to those parameters. Given a set of at least 2 points common to both datums, the set of observations for point *i* takes the form below.

$$\begin{bmatrix} X_{t,i} - X_{s,i} \\ Y_{t,i} - Y_{s,i} \\ Z_{t,i} - Z_{s,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X_{s,i} - X_m \\ 0 & 1 & 0 & Y_{s,i} - Y_m \\ 0 & 0 & 1 & Z_{s,i} - Z_m \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \Delta S \end{bmatrix} + \begin{bmatrix} v_{X,i} \\ v_{Y,i} \\ v_{Z,i} \end{bmatrix} \tag{4-23}$$

For *n* data points, this gives rise to 3*n* observation equations expressible in the form (4-1). Solution of (4-2) or application of (4-3) will produce the least-squares estimate of the parameters.

When the transformation involves local level coordinates, the parameters are those of the transformation X'_s, Y'_s and Z'_s to X'_t, Y'_t and Z'_t. They are derived in the same way as Helmert parameters. This covers what is described in Rapp (1993) and Leick (1995) as the Veis transformation. (See subsection 1.4.4 and Section 2.7.)

4.8 Simplified 7-parameter conformal transformation (Bursa-Wolf)

Unlike the rigorous 7PC model, the fully-linear Bursa-Wolf (FLBW) is linear with respect to its parameters, as shown in (2-41). It is therefore far more amenable to least-squares optimisation. When the parameters need to be derived from a set of points common to both datums, the method of least-squares optimisation can easily be applied to (2-41). This is one reason why Bursa-Wolf is often used in preference to the rigorous 7PC transformation. The validity of this approach depends on the rotations being small.

Given a set of at least 3 points common to both datums, the set of observations for point *i* takes the form below.

$$\begin{bmatrix} X_{t,i} - X_{s,i} \\ Y_{t,i} - Y_{s,i} \\ Z_{t,i} - Z_{s,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X_{s,i} & 0 & Z_{s,i} & -Y_{s,i} \\ 0 & 1 & 0 & Y_{s,i} & -Z_{s,i} & 0 & X_{s,i} \\ 0 & 0 & 1 & Z_{s,i} & Y_{s,i} & -X_{s,i} & 0 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \Delta S \\ R_X \\ R_Y \\ R_Z \end{bmatrix} + \begin{bmatrix} v_{X,i} \\ v_{Y,i} \\ v_{Z,i} \end{bmatrix} \quad (4\text{-}24)$$

For n data points, this gives rise to a full set of $3n$ observation equations that comply with (4-1). The least-squares estimate of the parameters can be obtained by solving (4-2) or applying (4-3).

The process recommended in Hofmann-Wellenhof and Moritz (2006) is to use Bursa-Wolf to derive the datum-shift parameters and the rigorous 7PC to apply them. This is what Varga *et al* (2017) appear to have done, since they link Bursa-Wolf to the matrix equation of the rigorous 7PC.

Iliffe and Lott (2008, page 98) advise that the derivation of a Bursa-Wolf transformation is ill-conditioned for small areas. There is "a high degree of correlation between the parameters" which, they argue, does not occur for Molodensky-Badekas. They recommend that Bursa-Wolf should only be used if the area subtends an angle of at least 30°.

Okwuashi and Eyoh (2012) present a similar argument about correlation. According to the quoted results, Molodensky-Badekas gives smaller residuals than Bursa-Wolf (ibid, Tables 2 and 3, notably the "LS" columns).

The claims in the above 2 paragraphs are untenable, for the following reasons. Equation (2-46) proves that FLBW and FLMB are mathematically equivalent. The parameters derived from Molodensky-Badekas are easily converted into parameters for Bursa-Wolf. The computed standard errors of the Bursa-Wolf translations might be bigger than those computed for the Molodensky-Badekas translations, but they must feed through to the computed coordinates in the same way. Equivalent equations are equally reliable if computed efficiently. The smaller residuals claimed in Okwuashi and Eyoh (2012) are at variance with the comparisons in European Union (2013) and Syetiawan *et al* (2019).

In parallel with the research undertaken for this thesis, Abbey and Featherstone (2020) rejected claims that Molodensky-Badekas is superior to Bursa-Wolf.

Advocates of Partially-Linear Bursa-Wolf have the problem that the transformation (2-42) does not appear to lead immediately to observation equations as straightforward as (4-24). Attempts to optimise PLBW have been published in Deakin (2006), Gacoki and Aduol (2002, Appendix 1) and Iliffe and Lott (2008, sub-appendix E.3.1). All of them introduce further approximations to linearise PLBW for optimisation purposes. In effect, all three treat the products ΔSR_X, ΔSR_Y and ΔSR_Z as negligible, which they usually are. Strictly speaking, the derived parameters optimise a slightly different transformation to the one that is going to be applied.

Every published PLBW optimisation method encountered during this study has made use of further approximations. These can be avoided, however, by a simple substitution based on (2-43). As noted in Section 2.8, this converts PLBW into FLBW. Ordinary least-squares can be used to optimise the parameters which include the FLBW versions of the rotations. The optimised PLBW rotations can then be computed exactly from

$$\begin{bmatrix} R_X^{PLBW} \\ R_Y^{PLBW} \\ R_Z^{PLBW} \end{bmatrix} = \frac{1}{1+\Delta S} \begin{bmatrix} R_X^{FLBW} \\ R_Y^{FLBW} \\ R_Z^{FLBW} \end{bmatrix}. \tag{4-25}$$

The parameters ΔX, ΔY, ΔZ and ΔS will be the same as they are not affected by the substitution.

Example: Bursa-Wolf derivation was applied to the dataset of 44 points in Great Britain given in sub-appendix C.2. The FLBW parameters obtained by ordinary least-squares optimisation were 445.181m, -161.834m, 542.616m, -0.732432″, 0.278998″, 1.607732″, -20.686319ppm. The optimal PLBW parameters are the same except for the rotations which were -0.732447″, 0.279003″, 1.607765″. These are 1/(1+ΔS) times the optimal FLBW rotations. In both cases, the 3D RMS residual is 2.5196m.

The choice of solution should depend on the version of Bursa-Wolf which is to be applied.

4.9 Simplified localised 7-parameter conformal transformation (Molodensky-Badekas)

Unlike the rigorous localised 7PC model, fully-linear Molodensky-Badekas (FLMB) is linear with respect to its parameters, as shown in (2-45). It is therefore far more amenable to least-squares optimisation. When the parameters need to be derived from a set of points common to both datums, the method of least-squares optimisation can easily be applied to (2-45). This is one reason why Molodensky-Badekas is often used in preference to the rigorous localised 7PC method.

Given a set of at least 3 points common to both datums, the set of observations for point *i* takes the form below.

$$\begin{bmatrix} X_{t,i} - X_{s,i} \\ Y_{t,i} - Y_{s,i} \\ Z_{t,i} - Z_{s,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X_{s,i} - X_m & 0 & Z_{s,i} - Z_m & -(Y_{s,i} - Y_m) \\ 0 & 1 & 0 & Y_{s,i} - Y_m & -(Z_{s,i} - Z_m) & 0 & X_{s,i} - X_m \\ 0 & 0 & 1 & Z_{s,i} - Z_m & Y_{s,i} - Y_m & -(X_{s,i} - X_m) & 0 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \Delta S \\ R_X \\ R_Y \\ R_Z \end{bmatrix} + \begin{bmatrix} v_{X,i} \\ v_{Y,i} \\ v_{Z,i} \end{bmatrix} \quad (4\text{-}26)$$

For *n* data points, this gives rise to a full set of 3*n* observation equations that comply with (4-1). The least-squares estimate of the parameters can be obtained by solving (4-2) or applying (4-3).

One option is to use Molodensky-Badekas to derive the datum-shift parameters and the rigorous 7PC transformation to apply them. This is what Varga *et al* (2017) appear to have done, since they link Molodensky-Badekas to the matrix equation of the rigorous localised 7PC method. The validity of this approach depends on the rotations being small.

In Section 2.9 it was noted that Deakin (2006), Iliffe and Lott (2008) and Kutoglu (2009a) are among the sources that prefer the PLMB given in (2-47). None of these mentions the simplest method of linearising the transformation for optimisation purposes.

That method is a simple substitution based on (2-48). As noted in Section 2.9, this converts PLMB into FLMB. Ordinary least-squares can be used to optimise the parameters which include the FLMB versions of the rotations. The optimised PLBW rotations can then be computed exactly from

$$\begin{bmatrix} R_X^{\mathrm{PLMB}} \\ R_Y^{\mathrm{PLMB}} \\ R_Z^{\mathrm{PLMB}} \end{bmatrix} = \frac{1}{1+\Delta S} \begin{bmatrix} R_X^{\mathrm{FLMB}} \\ R_Y^{\mathrm{FLMB}} \\ R_Z^{\mathrm{FLMB}} \end{bmatrix}. \quad (4\text{-}27)$$

The parameters ΔX, ΔY, ΔZ and ΔS will be the same as they are not affected by the substitution.

Example: Molodensky-Badekas derivation was applied to the dataset of 44 points in Great Britain given in sub-appendix C.2. The chosen local centroid was (370212.608, -157444.673, 5147839.809), calculated from the average of the OSGB36 Cartesian coordinates. The FLMB parameters obtained by ordinary least-squares optimisation were 376.414m, -111.300m, 431.635m, -0.732432″, 0.278998″, 1.607732″, -20.686319ppm. The optimal PLBW

parameters are the same except for the rotations which were -0.732447″, 0.279003″, 1.607765″. These are 1/(1+ΔS) times the optimal FLMB rotations. In both cases, the 3D RMS residual is 2.5196m.

The choice of solution should depend on the version of Molodensky-Badekas which is to be applied.

4.10 8-parameter affine transformation

Derivation of the parameters in this instance is hampered by the observation equations not being linear. For this study, a method has been devised to address the non-linearity, and hence derive the optimal parameters for this model. Because the method is more advanced than those covered in this Chapter, it is described in Section 5.3. The same section summarises the method in Andrei (2006).

4.11 9-parameter affine transformation

Deriving the best-fit parameters of this model is complicated by the non-linearity of the observation equations. For this study, a method has been devised to overcome the problem, and hence derive the optimal parameters for the chosen version (RS or SR) of this model. Since the method is more advanced than those covered in this Chapter, it is described in Section 5.4. The same section summarises the methods of other researchers.

4.12 12-parameter affine transformation

This method is linear with respect to its parameters, as shown by equation (2-58). Given a set of at least 5 points common to both datums, that form enables the parameters to be computed by least-squares optimisation.

4.13 Multiple regression equations and similar polynomial formulae

If datum shifts are modelled by polynomials of a pre-decided form (*ie* based on a particular set of monomials) then the parameters (*ie* the coefficients of the monomials) can be derived from linear observation equations using least-squares optimisation.

What distinguishes MREs from polynomial formulae in general is that they are only meant to include terms that are statistically significant. These are determined by a regression procedure. The most usual method is the "stepwise" multiple regression procedure described in Appelbaum (1982) and Section 7.2.4.3.3 of DMA (1987a). Least-squares optimisation is used

each time a variable is added (the choice being the variable that provides the greatest improvement in fitting the reference coordinate differences) and a statistical-significance test determines whether one or more variables should be removed.

This is a very high-level description of the process of determining which terms are included. In particular, it doesn't answer the question of how it decided which variables are added. Testing each possibility in turn makes for a very laborious process. In addition, the phrase "statistically significant" is a relative term.

The derivation of MREs is supposed to incorporate whatever distortions in the datum transformation are evident in the common points. The danger is that the polynomial components can introduce distortions of their own. The outermost control points therefore need to go as far as, or even slightly beyond, the area boundary. It is also important that the control points from which the polynomials are derived provide an even coverage of the area for which the MREs are to be applied.

Artificial control points are sometimes introduced to densify the data from which MREs are derived. Appelbaum (1982) used 53 points to derive a transformation from European Datum 1950 to WGS72. Only 33 points were Doppler stations. The other 20 points were located to provide more complete area coverage and the reference coordinate differences were obtained by interpolation (with some use made of Doppler stations external to the area).

Soycan (2005) used a different approach for deciding statistical significance in datum-shift models. In this adaptation, the models are *n*th-degree polynomials in the sense that they are linear combinations of monomials $u^i v^j$ for which $i + j \leq n$. "To determine the optimal polynomial degree, all possible alternatives (from degree one to five) have been considered for each solution, RMS values have been computed, polynomial coefficients have been tested, and only significant polynomial coefficients have been used." The resulting degree of polynomial turned out to be 2, both in the case of grid-coordinate shifts and in the case of Cartesian-coordinate shifts. 5th-degree monomials were therefore discarded *en bloc*, as were 4th-degree monomials and 3rd-degree monomials. Each shift, therefore, had precisely 6 terms (1, *u*, *v*, u^2, *uv* and v^2). One aspect of this approach is the likelihood that some of those 6 terms were less significant than some of the terms discarded.

The MRE-derivation method developed for this study is described in Section 5.5. It is intended for fully-normalised MREs, but could be adapted for MREs that use intermediate coordinates which are not fully normalised.

4.14 SMITSWAM

For Standard Molodensky in two stages with applied misclosure, the easiest way to compute the parameters is to use the process recommended in Section 4.3. SMITSWAM is, after all, equivalent to 3PC. It does require the geodetic coordinates of the control points to be converted to Cartesian coordinates, but that is a one-off exercise that needs no repetition once the optimal parameters are known.

4.15 Variations on Standard Molodensky transformation

The 7-parameter partially-conformal variation is considered below. The 6-parameter partially-conformal variation which omits R_Z is a straightforward modification (achieved by assuming R_Z is zero).

Given a set of at least 3 points common to both datums, the set of observations for point *i* takes the form below. As in Section 4.4, the radius of curvature in the prime vertical, as defined in equation (1-1), is denoted by *N* instead of ν.

$$\begin{aligned}(\rho_{s,i} + h_{s,i})\Delta\phi_i - (\Delta a/a_s)(N_{s,i}e_s^2 \sin\phi_{s,i}\cos\phi_{s,i}) \\ -\Delta f(\rho_{s,i}a_s/b_s + N_{s,i}b_s/a_s)\sin\phi_{s,i}\cos\phi_{s,i} = -\Delta X_{hor}\sin\phi_{s,i}\cos\lambda_{s,i} \\ -\Delta Y_{hor}\sin\phi_{s,i}\sin\lambda_{s,i} + \Delta Z_{hor}\cos\phi_{s,i} \quad + \quad v_{\phi,i}.\end{aligned} \tag{4-28}$$

$$(N_{s,i} + h_{s,i})\Delta\lambda_i\cos\phi_{s,i} = -\Delta X_{hor}\sin\lambda_{s,i} + \Delta Y_{hor}\cos\lambda_{s,i} + (N_{s,i} + h_{s,i})R_Z\cos\phi_{s,i} + v_{\lambda,i}. \tag{4-29}$$

$$\begin{aligned}\Delta h_i + \Delta a(a_s/N_{s,i}) - \Delta f(b_s/a_s)N_{s,i}\sin^2\phi_{s,i} = \Delta X_{ver}\cos\phi_{s,i}\cos\lambda_{s,i} \\ +\Delta Y_{ver}\cos\phi_{s,i}\,sin\,\lambda_{s,i} \; + \Delta Z_{ver}\sin\phi_{s,i} + v_{h,i}.\end{aligned} \tag{4-30}$$

Obtaining the 7 parameters ΔX_{hor}, ΔY_{hor}, ΔZ_{hor}, R_Z, ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} could be done by applying least-squares optimisation to $3n$ observations in 7 unknowns. Alternatively, two separate optimisations can be performed, so that

- ΔX_{hor}, ΔY_{hor}, ΔZ_{hor}, R_Z, are obtained from the $2n$ observation equations (4-28) and (4-29);
- ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} are obtained from the *n* observation equations (4-30).

In either case, the set of observations complies with (4-1). The least-squares estimate of the parameters can be obtained by solving (4-2) or applying (4-3).

Similarly, the 6-parameter variation can be obtained by either solving $3n$ equations in 6 unknowns or by obtaining $\Delta X_{hor}, \Delta Y_{hor}, \Delta Z_{hor}$ from the $2n$ "horizontal" observation equations and $\Delta X_{ver}, \Delta Y_{ver}, \Delta Z_{ver}$ from the n "vertical" observation equations. The values of $\Delta X_{hor}, \Delta Y_{hor}, \Delta Z_{hor}$ obtained from the "horizontal" equations can be cited as the values of ΔX, ΔY and ΔZ "that give Standard Molodensky the best horizontal fit". If this is done, there needs to be a caveat: "Do *not* use these values in the equation for Δh".

4.16 Variations on Abridged Molodensky transformation

It is sufficient to consider the 7-parameter partially-conformal variation; the 6-parameter case which assumes $R_Z = 0$ is a straightforward modification.

Given a set of at least 3 control points, the set of observations for point i takes the form below. As in Section 4.5, N rather than ν denotes the radius of curvature in the prime vertical.

$$\rho_{s,i}\Delta\phi_i - (a_s\Delta f + f_s\Delta a)\sin 2\phi_{s,i} = -\Delta X_{hor}\sin\phi_{s,i}\cos\lambda_{s,i} - \Delta Y_{hor}\sin\phi_{s,i}\sin\lambda_{s,i} + \Delta Z_{hor}\cos\phi_{s,i} \quad + \quad v_{\phi,i}. \tag{4-31}$$

$$N_{s,i}\Delta\lambda_t\cos\phi_{s,i} = -\Delta X_{hor}\sin\lambda_{s,i} + \Delta Y_{hor}\cos\lambda_{s,i} + R_Z N_{s,i}\cos\phi_{s,i} + v_{\lambda,i}. \tag{4-32}$$

$$\Delta h_i - (a_s\Delta f + f_s\Delta a)\sin^2\phi_{s,i} + \Delta a = \Delta X_{ver}\cos\phi_{s,i}\cos\lambda_{s,i} + \Delta Y_{ver}\cos\phi_{s,i}\sin\lambda_{s,i} + \Delta Z_{ver}\sin\phi_{s,i} \quad + \quad v_{h,i}. \tag{4-33}$$

The 7 parameters $\Delta X_{hor}, \Delta Y_{hor}, \Delta Z_{hor}, R_Z, \Delta X_{ver}, \Delta Y_{ver}, \Delta Z_{ver}$ could be obtained by applying least-squares optimisation to $3n$ observations in 7 unknowns. The alternative is two separate optimisations:

- $\Delta X_{hor}, \Delta Y_{hor}, \Delta Z_{hor}, R_Z$ are obtained from the $2n$ observation equations (4-31) and (4-32);
- $\Delta X_{ver}, \Delta Y_{ver}, \Delta Z_{ver}$ are obtained from the n observation equations (4-33).

In either case, the observations take the form given in (4-1); therefore, solution of (4-2) or application of (4-3) will produce the least-squares estimate of the parameters.

One thing to note about equations (4-31) to (4-33) is that – if the ρ and N terms are evaluated at the ellipsoid – equation (4-33) is the only one to make use of heights. It is therefore possible to derive $\Delta X_{hor}, \Delta Y_{hor}, \Delta Z_{hor}$ and R_Z purely from observation equations of the form (4-31) and (4-32), even if heights are unavailable. Obviously, the resulting transformation should only be used to compute horizontal datum shifts $\Delta\phi$ and $\Delta\lambda$.

Similarly, in the 6-parameter variation, it is possible to derive ΔX_{hor}, ΔY_{hor} and ΔZ_{hor} purely from observation equations of the form (4-31) and (4-32), even if heights are unavailable. The resulting values of ΔX_{hor}, ΔY_{hor}, ΔZ_{hor} can be cited as the values of ΔX, ΔY and ΔZ "that give Abridged Molodensky the best horizontal fit". If this is done, there needs to be a caveat: "Do *not* use these values in the equation for Δh".

4.17 Fully-normalised multiple regression equations (5 types)

Fully-normalised MREs are linear with respect to the parameters. For a pre-decided set of basis functions, the parameters are easily determined by least-squares optimisation. As with the MREs discussed in Sections 2.13 and 4.13, an analysis of statistical significance is required to decide which basis functions to include.

The statistical-significance analysis adopted for this study is described in Section 5.5.

CHAPTER 5: MORE ADVANCED TRANSFORMATION-DERIVATIONS

This chapter considers derivations of transformations that are non-linear with respect to the parameters. Most of the methods mentioned in Chapter 2 do not fall into this category, so there is no attempt to align Section numbers. Hence, for example, Section 5.1 is unrelated to Section 2.1.

This chapter also considers the derivation of transformations that take the form of multiple regression equations. These are linear with respect to the parameters and the models are easy to optimise for a given set of basis functions. However, deciding *which* possible combination of basis functions to use is an advanced topic in its own right.

5.1 Optimal Helmert transformation

As shown in Section 2.6, the 7PC (Helmert) model uses a rotation matrix that is non-linear with respect to the rotation parameters. This complicates the task of optimising Helmert parameters, because the least-squares methodology described in Chapter 4 requires a linear relationship.

The methods encountered during this study are as follows.

- An iterative least-squares optimisation process. Fang (2014) classifies one method as "quasi indirect errors adjustment (QIEA)" using a quasi-Newton (Broyden-Fletcher-Goldfarb-Shanno) solution. In that paper, the rotation matrix takes the form (2-15) or equivalently (2-16).
- The more direct method given by Awange and Grafarend (2002). An algebraic technique called "Groebner basis" is used to solve the problem when the rotation matrix takes the form (2-17) or equivalently (2-18). Hashemi *et al* (2013) also describes a method "using Gröbner bases techniques and solving polynomial systems", but it is unclear which form their rotation matrix takes.
- A "Procrustean solution" for N points given by Paláncz *et al* (2010) using a complicated algorithm from Gower and Dijksterhuis (2004).
- A novel RANSAC robust estimation technique described in Paláncz *et al* (2017), which allows for possible outliers among the data points. (RANSAC stands for random sample consensus.)
- Computation of approximate parameters from a small subset of the common points followed by linearisation of the observations equations based on those approximate

values. Least-squares optimisation is then applied to the linearised Helmert transformation model. The method is described and demonstrated in Fan (2005). The iterative phase is also described in Wolf and Ghilani (1997) which omits the issue of initial approximation.

All the sources quoted consider only one or other of the two commonly-used versions of Helmert (which were defined in subsections 2.6.1 and 2.6.2). This falls short of a general approach to derive either version.

It is also unclear whether these methods derive the optimal Helmert parameters. Awange and Grafarend (2002) includes a worked example based on the dataset in sub-appendix C.7.1. However, the derived transformation gives a worse fit to the control data than the method discussed in Section 4.6 (using Helmert with optimal Bursa-Wolf parameters).

The method devised for this study is believed to be original, although the third stage can be considered a variation on the iteration stage used by Fan (2005). It consists of the following processes, each involving linear least-squares optimisation. "HO" denotes Helmert optimisation.

- HO1: Distance analysis to obtain the original estimate of the scale change (ΔS_{DA}). In this case, only the possible impact of measurement errors might stop ΔS_{DA} being optimal because the 7PC transformation applies rotations exactly.
- HO2: Derivation of initial approximate translation and rotation parameters. This is done from what would be the partially-linear version of Bursa-Wolf except that the scale change has been already determined.
- HO3: Iteration using a re-linearisation of Helmert (with scale change ΔS fixed) based on corrections to the approximate rotation parameters; the minimised root-mean-square distance residual corresponding to the estimated scale change is denoted MinRMS(ΔS,7PC).
- HO4: Verification of optimality by repeating HO2 and HO3 for small deviations from ΔS_{DA}.

Stage HO1 is based on distances of the control points to the central point in both datums. The process is given by (5-1) to (5-5) below.

$$\left.\begin{array}{llll} X_{s,m} = \frac{1}{n}\sum_{i=1}^{n} X_{s,i}\,, & Y_{s,m} = \frac{1}{n}\sum_{i=1}^{n} Y_{s,i}\,, & Z_{s,m} = \frac{1}{n}\sum_{i=1}^{n} Z_{s,i}\,, \\ X_{t,m} = \frac{1}{n}\sum_{i=1}^{n} X_{t,i}\,, & Y_{t,m} = \frac{1}{n}\sum_{i=1}^{n} Y_{t,i}\,, & Z_{t,m} = \frac{1}{n}\sum_{i=1}^{n} Z_{t,i}\,. \end{array}\right\} \quad (5\text{-}1)$$

$$\left.\begin{aligned} d_{s,i} &= \sqrt{(X_{s,i} - X_{s,m})^2 + (Y_{s,i} - Y_{s,m})^2 + (Z_{s,i} - Z_{s,m})^2} \\ d_{t,i} &= \sqrt{(X_{t,i} - X_{t,m})^2 + (Y_{t,i} - Y_{t,m})^2 + (Z_{t,i} - Z_{t,m})^2} \end{aligned}\right\} \quad (5\text{-}2)$$

Treating S=1+ΔS as the parameter to be determined, the n observation equations take the form

$$d_{t,i} = S d_{s,i} + v_i \text{ or } d_{s,i} S + v_i = d_{t,i}. \quad (5\text{-}3)$$

These equations can be expressed in the form $\mathbf{Ax} + \mathbf{v} = \mathbf{b}$; since there is only one parameter in this stage, the design matrix **A** is simply the column vector containing the distances $d_{s,1}$ to $d_{s,n}$.

The normal equation for these observation equations takes the form

$$(\sum_{i=1}^{n} d_{s,i}^2) S = \sum_{i=1}^{n} d_{s,i} d_{t,i}. \quad (5\text{-}4)$$

The least-squares solution, which determines ΔS as well as S, is therefore

$$S_{\mathrm{DA}} = \sum_{i=1}^{n} d_{s,i} d_{t,i} \; / \sum_{i=1}^{n} d_{s,i}^2 \quad (5\text{-}5)$$

This completes stage HO1, but, two points are worth noting before proceeding further.

Firstly, the idea of using distance analysis to approximate S has been considered in the following papers.

- Schut (1973) which gives no formula for the initial approximation.
- Paláncz *et al* (2010) which uses the ratio $(\sum d_{t,i})/(\sum d_{s,i})$.
- Han (2010) which uses the mean of the quotients of distances between data points in the target datum over corresponding distances in the source datum. Han's approach to parameter computation is, however, not based on least-squares.
- Závoti and Kalmár (2016) which suggests three possible expressions for S. Using the notation of this Section, they are $(\sum d_{t,i})/(\sum d_{s,i})$, $\sqrt{(\sum d_{t,i}^2)/(\sum d_{s,i}^2)}$ and $(\sum d_{s,i} d_{t,i})/(\sum d_{s,i}^2)$. The paper discusses when those expressions are equivalent rather than advocating the third one.

Secondly, application of (4-6) to (5-3) gives the following result:

$$\text{Standard error of } S_{\mathrm{DA}} = \sqrt{\frac{\sum_{i=1}^{n} v_i^2}{(n-1)\sum_{i=1}^{n} d_i^2}} \quad (5\text{-}6)$$

If this quantity is larger than the difference between S_{DA} and 1, then the scale change is statistically insignificant and there is a strong case for setting S to 1.

Stage HO2 uses the value of *S* obtained by least-squares.

Regarding *S* as already optimised, the Helmert equation can be written in the form

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{R} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix} \tag{5-7}$$

or

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix} \tag{5-8}$$

where the matrix elements $r_{i,j}$ are non-linear functions of the rotations R_X, R_Y and R_Z. This is true whether the rotation matrix takes the form (2-16) or (2-18).

In the context of geodetic datum transformations, rotations are generally smaller than 1° (considerably smaller, in fact). That makes their sines approximately their value in radians, their cosines approximately 1, and the products of their sines negligible. Whether the rotation matrix is given by (2-16) or (2-18), it can be approximated as follows.

$$\begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix} \cong \begin{bmatrix} 1 & -R_Z & R_Y \\ R_Z & 1 & -R_X \\ -R_Y & R_X & 1 \end{bmatrix}. \tag{5-9}$$

Substituting (5-9) into (5-8),

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} \cong \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} 1 & -R_Z & R_Y \\ R_Z & 1 & -R_X \\ -R_Y & R_X & 1 \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix} \tag{5-10}$$

and hence

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} \cong \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & SZ_s & -SY_s \\ 0 & 1 & 0 & -SZ_s & 0 & SX_s \\ 0 & 0 & 1 & SY_s & -SX_s & 0 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ R_X \\ R_Y \\ R_Z \end{bmatrix}. \tag{5-11}$$

The observation equations take the form

$$\begin{bmatrix} X_{t,i} - SX_{s,i} \\ Y_{t,i} - SY_{s,i} \\ Z_{t,i} - SZ_{s,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & SZ_{s,i} & -SY_{s,i} \\ 0 & 1 & 0 & -SZ_{s,i} & 0 & SX_{s,i} \\ 0 & 0 & 1 & SY_{s,i} & -SX_{s,i} & 0 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ R_X \\ R_Y \\ R_Z \end{bmatrix} + \begin{bmatrix} v_{X,i} \\ v_{Y,i} \\ v_{Z,i} \end{bmatrix}. \tag{5-12}$$

Since this complies with (4-1), the least-squares estimate of the 6 unknowns can be obtained by solving (4-2) or applying (4-3). This will provide starting approximations $\bar{R}_X, \bar{R}_Y, \bar{R}_Z$ to the rotation parameters.

Stage HO3 begins with the substitution

$$\begin{bmatrix} R_X \\ R_Y \\ R_Z \end{bmatrix} = \begin{bmatrix} \bar{R}_X \\ \bar{R}_Y \\ \bar{R}_Z \end{bmatrix} + \begin{bmatrix} \delta R_X \\ \delta R_Y \\ \delta R_Z \end{bmatrix}. \tag{5-13}$$

Then each term in the rotation matrix **R** can be expressed in the form

$$r_{i,j} = \bar{r}_{i,j} + \delta R_X(\partial r_{i,j}/\partial R_X) + \delta R_Y(\partial r_{i,j}/\partial R_Y) + \delta R_Z(\partial r_{i,j}/\partial R_Z) \tag{5-14}$$

where the partial derivatives are evaluated at $\bar{R}_X, \bar{R}_Y, \bar{R}_Z$. The overbar in $\bar{r}_{i,j}$ denotes $r_{i,j}$ evaluated at $\bar{R}_X, \bar{R}_Y, \bar{R}_Z$.

This enables equation (5-8) to be rewritten in the form

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix} + \begin{bmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,1} & m_{3,2} & m_{3,3} \end{bmatrix} \begin{bmatrix} \delta R_X \\ \delta R_Y \\ \delta R_Z \end{bmatrix} \tag{5-15}$$

where the terms $m_{i,j}$ can be evaluated from the following equations.

$$m_{i,1} = (\partial r_{i,1}/\partial R_X)SX_s + (\partial r_{i,2}/\partial R_X)SY_s + (\partial r_{i,3}/\partial R_X)SZ_s; \tag{5-16}$$

$$m_{i,2} = (\partial r_{i,1}/\partial R_Y)SX_s + (\partial r_{i,2}/\partial R_Y)SY_s + (\partial r_{i,3}/\partial R_Y)SZ_s; \tag{5-17}$$

$$m_{i,3} = (\partial r_{i,1}/\partial R_Z)SX_s + (\partial r_{i,2}/\partial R_Z)SY_s + (\partial r_{i,3}/\partial R_Z)SZ_s. \tag{5-18}$$

Equations (5-16) to (5-18) can be expressed in vector form:

$$\begin{bmatrix} m_{1,1} \\ m_{2,1} \\ m_{3,1} \end{bmatrix} = \begin{bmatrix} \partial r_{1,1}/\partial R_X & \partial r_{1,2}/\partial R_X & \partial r_{1,3}/\partial R_X \\ \partial r_{2,1}/\partial R_X & \partial r_{2,2}/\partial R_X & \partial r_{2,3}/\partial R_x \\ \partial r_{3,1}/\partial R_X & \partial r_{3,2}/\partial R_X & \partial r_{3,3}/\partial R_X \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-19}$$

$$\begin{bmatrix} m_{1,2} \\ m_{2,2} \\ m_{3,2} \end{bmatrix} = \begin{bmatrix} \partial r_{1,1}/\partial R_Y & \partial r_{1,2}/\partial R_Y & \partial r_{1,3}/\partial R_Y \\ \partial r_{2,1}/\partial R_Y & \partial r_{2,2}/\partial R_Y & \partial r_{2,3}/\partial R_Y \\ \partial r_{3,1}/\partial R_Y & \partial r_{3,2}/\partial R_Y & \partial r_{3,3}/\partial R_Y \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-20}$$

$$\begin{bmatrix} m_{1,3} \\ m_{2,3} \\ m_{3,3} \end{bmatrix} = \begin{bmatrix} \partial r_{1,1}/\partial R_Z & \partial r_{1,2}/\partial R_Z & \partial r_{1,3}/\partial R_Z \\ \partial r_{2,1}/\partial R_Z & \partial r_{2,2}/\partial R_Z & \partial r_{2,3}/\partial R_Z \\ \partial r_{3,1}/\partial R_Z & \partial r_{3,2}/\partial R_Z & \partial r_{3,3}/\partial R_Z \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-21}$$

Equation (5-15) can be expressed in terms of linear combinations of ΔX, ΔY, ΔZ, δR_X, δR_Y and δR_Z.

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} - \begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & m_{1,1} & m_{1,2} & m_{1,3} \\ 0 & 1 & 0 & m_{2,1} & m_{2,2} & m_{2,3} \\ 0 & 0 & 1 & m_{3,1} & m_{3,2} & m_{3,3} \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \delta R_X \\ \delta R_Y \\ \delta R_Z \end{bmatrix} \tag{5-22}$$

For the *i*th point, the observation equations take the form shown below.

$$\begin{bmatrix} X_{t,i} \\ Y_{t,i} \\ Z_{t,i} \end{bmatrix} - \begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} \begin{bmatrix} SX_{s,i} \\ SY_{s,i} \\ SZ_{s,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & m_{1,1,i} & m_{1,2,i} & m_{1,3,i} \\ 0 & 1 & 0 & m_{2,1,i} & m_{2,2,i} & m_{2,3,i} \\ 0 & 0 & 1 & m_{3,1,i} & m_{3,2,i} & m_{3,3,i} \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \delta R_X \\ \delta R_Y \\ \delta R_Z \end{bmatrix} + \begin{bmatrix} v_{X,i} \\ v_{Y,i} \\ v_{Z,i} \end{bmatrix}. \tag{5-23}$$

Since this complies with (4-1), the least-squares estimate of the 6 unknowns can be obtained by solving (4-2) or applying (4-3). The values of δR_X, δR_Y and δR_Z are added to $\bar{R}_X$, $\bar{R}_Y$, $\bar{R}_Z$ to obtain new approximations to the rotation parameters. The iteration step is repeated until convergence.

All that remains is to provide explicit formulae for the matrix elements in equation (5-22).

Consider the case of Version-1 Helmert, where the rotation matrix satisfies equation (2-16). Using $\bar{c}_X$ to denote $\cos\bar{R}_X$, $\bar{s}_X$ to denote $\sin\bar{R}_X$, etc, the definition of $\bar{r}_{i,j}$ ensures that

$$\begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} = \begin{bmatrix} \bar{c}_Y\bar{c}_Z & \bar{s}_X\bar{s}_Y\bar{c}_Z - \bar{c}_X\bar{s}_Z & \bar{s}_X\bar{s}_Z + \bar{c}_X\bar{s}_Y\bar{c}_Z \\ \bar{c}_Y\bar{s}_Z & \bar{c}_X\bar{c}_Z + \bar{s}_X\bar{s}_Y\bar{s}_Z & \bar{c}_X\bar{s}_Y\bar{s}_Z - \bar{s}_X\bar{c}_Z \\ -\bar{s}_Y & \bar{s}_X\bar{c}_Y & \bar{c}_X\bar{c}_Y \end{bmatrix}. \tag{5-24}$$

From (5-19),

$$\begin{bmatrix} m_{1,1} \\ m_{2,1} \\ m_{3,1} \end{bmatrix} = \begin{bmatrix} 0 & \bar{c}_X\bar{s}_Y\bar{c}_Z + \bar{s}_X\bar{s}_Z & \bar{c}_X\bar{s}_Z - \bar{s}_X\bar{s}_Y\bar{c}_Z \\ 0 & \bar{c}_X\bar{s}_Y\bar{s}_Z - \bar{s}_X\bar{c}_Z & -(\bar{s}_X\bar{s}_Y\bar{s}_Z + \bar{c}_X\bar{c}_Z) \\ 0 & \bar{c}_X\bar{c}_Y & -\bar{s}_X\bar{c}_Y \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-25}$$

From (5-20),

$$\begin{bmatrix} m_{1,2} \\ m_{2,2} \\ m_{3,2} \end{bmatrix} = \begin{bmatrix} -\bar{s}_Y\bar{c}_Z & \bar{s}_X\bar{c}_Y\bar{c}_Z & \bar{c}_X\bar{c}_Y\bar{c}_Z \\ -\bar{s}_Y\bar{s}_Z & \bar{s}_X\bar{c}_Y\bar{s}_Z & \bar{c}_X\bar{c}_Y\bar{s}_Z \\ -\bar{c}_Y & -\bar{s}_X\bar{s}_Y & -\bar{c}_X\bar{s}_Y \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-26}$$

From (5-21),

$$\begin{bmatrix} m_{1,3} \\ m_{2,3} \\ m_{3,3} \end{bmatrix} = \begin{bmatrix} -\bar{c}_Y\bar{s}_Z & -(\bar{s}_X\bar{s}_Y\bar{s}_Z + \bar{c}_X\bar{c}_Z) & \bar{s}_X\bar{c}_Z - \bar{c}_X\bar{s}_Y\bar{s}_Z \\ \bar{c}_Y\bar{c}_Z & \bar{s}_X\bar{s}_Y\bar{c}_Z - \bar{c}_X\bar{s}_Z & \bar{c}_X\bar{s}_Y\bar{c}_Z + \bar{s}_X\bar{s}_Z \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-27}$$

Consider the case of Version-2 Helmert, where the rotation matrix satisfies equation (2-18). Using $\bar{c}_X$ to denote $\cos\bar{R}_X$, $\bar{s}_X$ to denote $\sin\bar{R}_X$, etc, the definition of $\bar{r}_{i,j}$ ensures that

$$\begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} = \begin{bmatrix} \bar{c}_Y\bar{c}_Z & -\bar{c}_Y\bar{s}_Z & \bar{s}_Y \\ \bar{s}_X\bar{s}_Y\bar{c}_Z + \bar{c}_X\bar{s}_Z & \bar{c}_X\bar{c}_Z - \bar{s}_X\bar{s}_Y\bar{s}_Z & -\bar{s}_X\bar{c}_Y \\ \bar{s}_X\bar{s}_Z - \bar{c}_X\bar{s}_Y\bar{c}_Z & \bar{c}_X\bar{s}_Y\bar{s}_Z + \bar{s}_X\bar{c}_Z & \bar{c}_X\bar{c}_Y \end{bmatrix} \tag{5-28}$$

From (5-19),

$$\begin{bmatrix} m_{1,1} \\ m_{2,1} \\ m_{3,1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ \bar{c}_X\bar{s}_Y\bar{c}_Z - \bar{s}_X\bar{s}_Z & -(\bar{s}_X\bar{c}_Z + \bar{c}_X\bar{s}_Y\bar{s}_Z) & -\bar{c}_X\bar{c}_Y \\ \bar{c}_X\bar{s}_Z + \bar{s}_X\bar{s}_Y\bar{c}_Z & \bar{c}_X\bar{c}_Z - \bar{s}_X\bar{s}_Y\bar{s}_Z & -\bar{s}_X\bar{c}_Y \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-29}$$

From (5-20),

$$\begin{bmatrix} m_{1,2} \\ m_{2,2} \\ m_{3,2} \end{bmatrix} = \begin{bmatrix} -\bar{s}_Y\bar{c}_Z & \bar{s}_Y\bar{s}_Z & \bar{c}_Y \\ \bar{s}_X\bar{c}_Y\bar{c}_Z & -\bar{s}_X\bar{c}_Y\bar{s}_Z & \bar{s}_X\bar{s}_Y \\ -\bar{c}_X\bar{c}_Y\bar{c}_Z & \bar{c}_X\bar{c}_Y\bar{s}_Z & -\bar{c}_X\bar{s}_Y \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-30}$$

From (5-21),

$$\begin{bmatrix} m_{1,3} \\ m_{2,3} \\ m_{3,3} \end{bmatrix} = \begin{bmatrix} -\bar{c}_Y\bar{s}_Z & -\bar{c}_Y\bar{c}_Z & 0 \\ \bar{c}_X\bar{c}_Z - \bar{s}_X\bar{s}_Y\bar{s}_Z & -(\bar{c}_X\bar{s}_Z + \bar{s}_X\bar{s}_Y\bar{c}_Z) & 0 \\ \bar{s}_X\bar{c}_Z + \bar{c}_X\bar{s}_Y\bar{s}_Z & \bar{c}_X\bar{s}_Y\bar{c}_Z - \bar{s}_X\bar{s}_Z & 0 \end{bmatrix} \begin{bmatrix} SX_s \\ SY_s \\ SZ_s \end{bmatrix}. \tag{5-31}$$

Collectively, stages HO2 and HO3 derive a rigid transformation. For this study, an Excel VBA subroutine was written to perform this derivation. It can be found in sub-appendix G.2.4.

Stage HO4 involves repetition of the stages HO2 and HO3 for two slightly different scale changes, $\Delta S_{\mathrm{DA}} - \delta S$ and $\Delta S_{\mathrm{DA}} + \delta S$. The residuals enable the computation of three values of MinRMS(Δ*S*, 7PC). If the latter is evaluated to 8 or 9 decimal places, 0.01ppm would be a suitable value for δS.

If MinRMS is less for the distance-analysis estimate of ΔS than for the two neighbouring estimates, and the outer values of MinRMS are closer to each other than to the central value, then the distance-analysis estimate of ΔS can be regarded as optimal along with the corresponding translation and rotation parameters obtained by stages HO2 and HO3. The condition is illustrated in Figure 5-1, in which MinRMS is near-symmetric about its central value.

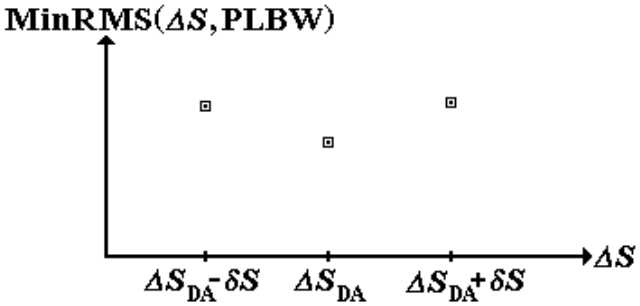


Figure 5-1: The pattern of minimum RMS distance residuals from the 7PC transformation that indicates that the ΔS obtained from distance analysis is the optimum value.

If the near-symmetry condition is not satisfied, then a quadratic curve can be fitted to the three values of (Δ*S*, MinRMS), as in Figure 5-2.

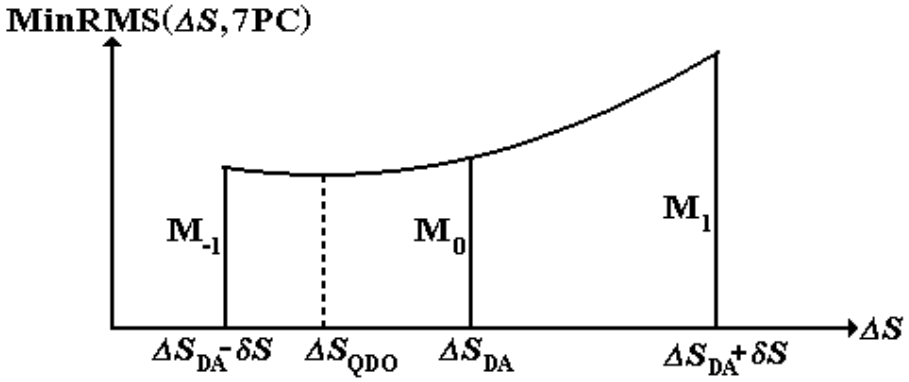


Figure 5-2: The quadratically-derived optimal ΔS based on three minimum RMS distance residuals from the 7PC transformation.

The value of ΔS that matches the minimum point on the curve is a quadratically-derived optimal value and can be denoted ΔS_{QDO}. The values of MinRMS are denoted by M_{-1}, M_0 and M_1. The curve does not need to be plotted, since ΔS_{QDO} can be computed from

$$\Delta S_{\mathrm{QDO}} = \Delta S_{\mathrm{DA}} + \frac{(M_{-1}-M_1)\delta S}{2(M_{-1}+M_1-2M_0)}. \tag{5-32}$$

Stages HO2 and HO3 are performed a 4th time to obtain the remaining parameters.

The use of formula (5-32) assumes that MinRMS behaves like a quadratic function around the optimal value of ΔS. All the tests carried out in this study confirm that assumption. To give one example, for the dataset in sub-appendix C.2, MinRMS had a minimum value (2.519810834m) when ΔS was -20.6863ppm. For i in the range -8 to 8, MinRMS(ΔS, 7PC) was almost indistinguishable from $(2.519642890 + 0.0000026244i^2)$m when $\Delta S = (-20.6863 \pm i/100)$ppm.

The four-stage method was applied to find the optimal Helmert Version 1 parameters for the transformations listed below (cross-referenced to Appendices C and D).

- AGD84 to GDA94 in Western Australia using 82 data points (C.1).
- OSGB36 to WGS84 in Great Britain using 44 data points (C.2).
- Accra to WGS84 in Ghana's Golden Triangle using 19 data points (C.3).
- SWEREF93 to RT90/RH70 in Sweden using 20 data points (C.4).
- Krassovsky to WGS84 (3 simulations) in "Helmatan" using 12 data points (D.1).
- SAD69 to WGS84 (3 simulations) in "Helmatia" using 16 data points (D.2).
- Arc 1950 to WGS84 (3 simulations) in "Helmatto" using 14 data points (D.3).
- Réunion to WGS84 (3 simulations) in "Helmatrun" using 13 data points (D.4).

- International Ellipsoid to WGS84 (3 simulations) in "St Fuitioci" using 12 data points (D.5).
- Bessel 1841 to WGS84 (3 simulations) in "Main Gyria" using 20 data points (D6).

The transformation parameters obtained are given in Table 5-1.

Table 5-1: Optimal Version-1 parameters derived from actual and simulated datasets

Dataset	ΔX (m)	ΔY (m)	ΔZ (m)	ΔS (ppm)	$R_X('')$	$R_Y('')$	$R_Z('')$
C.1	-115.83771	-48.37321	144.75955	3.68981458	0.11971214	0.38398838	0.37039627
C.2	445.18103	-161.83410	542.61595	-20.68629118	-0.73244160	0.27900550	1.60776264
C.3	-151.19021	31.59316	327.17659	-7.16772580	-0.44517945	0.00581813	-0.02199526
C.4	-419.56843	-99.24597	-591.45587	1.02365275	-0.85018849	-1.81414510	7.85347921
D.1 (Sim 1)	-25.97175	66.96656	-215.58018	7.88825821	6.88237925	-6.80419106	-11.56950830
D.1 (Sim 2)	-25.97165	66.96753	-215.58083	17.89830362	16.12371323	-16.16292380	-21.56056403
D.1 (Sim 3)	-25.97112	66.96956	-215.58215	22.44843559	22.72662595	-21.91994919	-42.65174832
D.2 (Sim 1)	-50.63311	-130.97910	90.47817	-15.18944085	9.56675112	-8.37521909	14.56740251
D.2 (Sim 2)	-50.63323	-130.97865	90.47870	-16.68945726	21.08555751	-15.75695369	23.50618429
D.2 (Sim 3)	-50.63320	-130.97800	90.47900	-18.18951039	25.88288600	-22.31974602	41.38499204
D.3 (Sim 1)	-77.21314	-103.53758	79.53254	-12.97571104	10.39471472	-8.12729717	-14.08503079
D.3 (Sim 2)	-77.21314	-103.53790	79.52175	-14.57541748	21.67586939	-15.23604061	-22.24613350
D.3 (Sim 3)	-77.21234	-103.53908	79.51169	-16.17514328	26.37833112	-21.55381994	-38.56716019
D.4 (Sim 1)	775.31588	-608.85589	-62.29840	-32.33400211	-0.36612789	77.00092571	-11.20496525
D.4 (Sim 2)	775.31090	-608.85392	-62.29993	-28.81949018	-0.60357050	47.03090681	-6.95515844
D.4 (Sim 3)	775.31588	-608.85589	-62.29840	-32.33400211	-0.36612789	77.00092571	-11.20496525
D.5 (Sim 1)	1159.21272	-2047.20929	917.63820	200.75920949	42.57768970	-93.33285762	38.16074892
D.5 (Sim 2)	959.21544	-1097.20146	317.45958	80.74850000	25.58961200	-50.33217866	30.17607838
D.5 (Sim 3)	959.21529	-879.19931	317.40055	47.74470000	20.59296124	-37.33210865	26.17918694
D.6 (Sim 1)	214.41329	-155.60150	-221.77581	154.33300000	16.83221351	-63.96604668	32.89380878
D.6 (Sim 2)	214.39760	178.39256	-221.79482	94.33300000	15.83754804	-41.96554491	25.89564848
D.6 (Sim 3)	214.38912	407.39446	-221.80495	64.33200000	14.83990228	-30.96517251	20.89659788

The first significant finding from these tests was that stage HO3 only needed a single iteration; the second iteration merely confirmed convergence. On this evidence, the re-linearisation only needs to be applied once.

The second significant finding was that the "distance-analysis" scale-change proved to be optimal, except in the case of three simulated datasets for which the area of coverage was significantly smaller than a degree square.

The exceptional cases were the three simulations for St Fuitioci. They involved very different rotations and scale changes (see last 3 rows of above table) but they had the same "measurement errors" (perturbations to coordinates). Stage HO4 showed that the optimal ΔS differed from ΔS_{DA} by 0.04ppm. Halving the simulated measurement errors reduced the difference to 0.01ppm. This suggested that small areas increase the size of measurement errors relative to the inter-point distances, undermining the validity of ΔS_{DA} as a scale-change estimate (albeit slightly). This explanation was confirmed when doubling of the measurement

errors and doubling of all inter-point distances had a totally neutral effect on the difference between ΔS_{DA} and the optimal ΔS.

In the St Fuitioci simulations, the original ΔS was near-optimal in the sense that

$$\mathrm{MinRMS}(\Delta S_{\mathrm{opt}}, 7PC) = \mathrm{MinRMS}(\Delta S_{\mathrm{DA}}, 7PC) - 0.0003\mathrm{mm}. \tag{5-33}$$

The differences in transformed coordinates were at the sub-millimetre level (just). Even so, the use of a quadratically-derived optimal ΔS proved effective in optimising ΔS.

In practice, therefore, the Helmert optimisation process consists of:

- HO1: Distance analysis to obtain the original estimate of the scale change (ΔS_{DA}).
- HO2: Derivation of initial approximate translation and rotation parameters. This is done from what would be the partially-linear version of Bursa-Wolf except that the scale change has been already determined. This stage applies least-squares to the observation equations (5-12).
- HO3: Deriving improved translation and rotation parameters using a re-linearisation of Helmert (with scale change ΔS fixed) based on corrections to the approximate rotation parameters; further iteration will be unnecessary (except possibly when testing the software). Least-squares optimisation is applied to the observation equations (5-23).
- HO4: Verification of optimality by repeating HO2 and HO3 for small deviations from ΔS_{DA}. This is only necessary when testing the software for stages HO1 to HO3, or in cases where all control points are within 150 kilometres of each other.

This method is also described in Ruffhead (2021a), which is based on the first two sections of this chapter.

5.2 Pseudo-optimal Helmert transformation

Derivation of either version of Helmert requires control points. In most cases they are not publicly available. Proof of their existence can be found in tables of parameters of Bursa-Wolf transformations, notably Table 5 of ESRI (2012).

It is not possible to derive the optimal Version 1 of Helmert purely from the optimised Bursa-Wolf. However, it is possible to derive a close-fitting "pseudo-optimal" Helmert transformation by

- applying the totally-linear Bursa-Wolf model to 25-50 points that give good coverage of the geographical region in question, and
- deriving the Helmert transformation which gives the best fit at these virtual data points. This can be done by the stages HO1, HO2 and HO3 (without further iteration) described in Section 5.1.

If the RMS fit of Helmert to the virtual points is small compared with the RMS fit of Bursa-Wolf to the original control points, then the pseudo-optimal Helmert transformation should give almost as good a fit as the optimal Bursa-Wolf. It can be regarded as the conformal model that comes closest to the optimal Bursa-Wolf. It will also provide an indication of how far the optimal Bursa-Wolf is to being conformal.

The smaller the rotation and scaling parameters, the closer Helmert is to Bursa-Wolf. Conversely the divergence between Helmert and Bursa-Wolf is likely to be greatest when the scale change and at least one rotation are large. Two notable examples are the following Bursa-Wolf transformations from Table 5 of ESRI (2012):

- Reunion 1947 To RGR 1992 for which the Bursa Wolf parameters are 789.524m, -626.486m, -89.904m, -32.3241ppm, 0.6006″, 76.7946″, -10.5788″.
- Fatu Iva 1972 To WGS84 for which the Bursa-Wolf parameters 347.103m, 1078.125m, 2623.922m, 186.074ppm, -33.8875″, 70.6773″, -9.3943″. The rotations have been converted to position-vector rotations.

Datasets of virtual points have been created for this study. They are shown in Appendix E, with diagrams of Réunion Island and Fatu Iva (sometimes spelt Fatu Hiva although the "H" is silent).

In each case, the method of deriving Helmert Version 1 was that described in Section 5.1. The convergence was rapid as in the tests in Section 5.1. The verification stage produced agreement with Figure 5-1 (unsurprisingly as there weren't any simulated or actual measurement errors). The transformation parameters obtained are tabulated below:

Dataset	ΔX (m)	ΔY (m)	ΔZ (m)	ΔS (ppm)	R_X (″)	R_Y (″)	R_Z (″)
E.1 (s→t)	789.70880	-626.93585	-89.93390	-32.26312476	0.60126857	76.79736169	-10.57263204
E.2 (s→t)	346.90967	1078.23235	2623.87087	186.1299981	-33.88457022	70.66260075	-9.395414631

The method of Section 3.6.1 was used to derive the Helmert Version 1 parameters the other way (target to source). The transformation parameters obtained are tabulated below:

Dataset	ΔX (m)	ΔY (m)	ΔZ (m)	ΔS (ppm)	$R_X('')$	$R_Y('')$	$R_Z('')$
E.1 (t→s)	-789.79985	626.91586	89.64092	32.26416570	-0.60520506	-76.79733077	10.57285664
E.2 (t→s)	-345.89726	-1077.61650	-2623.67829	-186.0953602	33.88135347	-70.66414317	9.38380681

These results could be put into a table of parameters for the following Helmert Version 1 transformations:

Reunion 1947 To RGR 1992

Fatu Iva 1972 To WGS84

RGR 1992 To Reunion 1947

WGS84 To Fatu Iva 1972

5.3 Derivation of 8-parameter affine transformation

The 8-parameter affine model in Section 2.10 uses a multiplicative matrix **RS** that is non-linear with respect to the rotation and scaling parameters. The task of optimising parameters therefore requires something more elaborate than the methodology described in Chapter 4 (which is for linear models).

Section 2.10 emphasised that the transformation should only be applied to local level coordinates in the two datums. It follows that the model is derived from local level coordinates in the source and target datums. The local level coordinates are derived from equations (2-49) and (2-50).

Andrei (2006) describes a method of obtaining the transformation (RS version) from a dataset of at least 3 control points. It requires initial approximations of the scale and rotations, but Andrei says nothing about how they are obtained. Corrections are obtained by least-squares from a linearisation of the observation equations. It is an iterative process and the algorithm (ibid, Appendix B) uses MATLAB software.

Andrei's algorithm in its most general form optimises 9 parameters, including 3 different scale factors. However, it uses a parameter "no_param" which is 8 in the case where two scale factors are equal. This case is treated as a linear constraint $S_1 - S_2 = 0$ and it is fed into the least-squares optimisation process known variously as "combined least-squares" (Cross, 1983) and "inclusion of infallible observations" (Bomford, 1980). Although this is statistically valid, it would have been far simpler to introduce the single horizontal scale factor by substitution in the 9-parameter algorithm.

One advantage of using ENU coordinates is that a third of the observation equations have residuals which are roughly vertical. This provides a means whereby height observations can be given lower weights than horizontal observations in the derivation of parameters. Details of this approach are given in Reit (1998).

The new method of deriving the optimal 8-parameter transformation introduced for this study is based on comparisons with the optimal 7PC model. The method for obtaining the latter is described in Section 5.1 and can easily be applied to the ENU coordinate system. Crucially, the scale factor is derived from distance analysis, giving ΔS_{DA}.

For this analysis, d_v and d_h will denote "vertical" and "horizontal" components of distance from a data point to the central point in the source datum. This is consistent with the use of S_v and S_h to denote the scale factors, as introduced in Section 2.10.

The key to the new method is the equivalent-enlargement hypothesis (EEH). This is based on the assumption that the 8-parameter affine transformation that best fits the coordinate data will have the same element of enlargement (or shrinking) as the best-fitting 7PC model.

$$\sum(S_v d_v)^2 + \sum(S_h d_h)^2 = \sum\left(S_{\mathrm{DA}}\sqrt{d_v^2 + d_h^2}\right)^2. \tag{5-34}$$

This may be written as

$$S_v^2\sum d_v^2 + S_h^2\sum d_h^2 = S_{\mathrm{DA}}^2\sum(d_v^2 + d_h^2). \tag{5-35}$$

Condition (5-35) enables ΔS_h to be expressed in terms of ΔS_v.

$$S_h = \sqrt{\frac{S_{\mathrm{DA}}^2\sum(d_v^2+d_h^2) - S_v^2\sum d_v^2}{\sum d_h^2}}. \tag{5-36}$$

In this approach, MinRMS(ΔS_v) is defined as the minimised RMS of the distance residual of the 8-parameter affine transformation resulting from a given choice of ΔS_v. This ΔS_v and the ΔS_h resulting from (5-36) are fed into equation (5-37) to obtain scaled coordinates.

$$\begin{bmatrix} X'_{s,i,sc} \\ Y'_{s,i,sc} \\ Z'_{s,i,sc} \end{bmatrix} = \begin{bmatrix} S_h & 0 & 0 \\ 0 & S_h & 0 \\ 0 & 0 & S_v \end{bmatrix} \begin{bmatrix} X'_{s,i} \\ Y'_{s,i} \\ Z'_{s,i} \end{bmatrix}. \tag{5-37}$$

The other 6 parameters (translations and rotations) are derived by applying stages HO2 & HO3 (described in Section 5.1) to the scaled source coordinates and target coordinates. For this

study, an Excel VBA subroutine was written to perform those stages; it is given in sub-appendix G.2.4.

The obvious starting value of ΔS_v is ΔS_{DA}, and MinRMS(ΔS_{DA}) is known from the Helmert optimisation. If MinRMS(ΔS_{DA}+0.1ppm) < MinRMS(ΔS_{DA}) then the appropriate search direction will be values of ΔS_v increasing from ΔS_{DA}. If not, then the appropriate search direction will be values of ΔS_v decreasing from ΔS_{DA}. The search process is concluded when the value of ΔS_v that minimises MinRMS is computed.

The new method was applied to find the optimal 8-parameter affine transformation for each of the following datasets (cross-referenced to Appendix C). In each case the parameters obtained were successfully tested by computing the RMS residual distance for scale factors around those obtained by the algorithm. The local level coordinates were obtained using Andrei's process as described in Section 2.10. (Six decimal places are shown merely to quantify the agreement between the "RS" and "SR" versions.)

- AGD84 to GDA94 in Western Australia using 82 data points (C.1). The RMS residual distance was 0.758246m for both the "RS" and "SR" versions.
- OSGB36 to WGS84 in Great Britain using 44 data points (C.2). The RMS residual distance was 2.504814m for the "RS" version and 2.504813m for the "SR" version.
- Accra to WGS84 in Ghana's Golden Triangle using 19 data points (C.3). The RMS residual distance was 0.961925m for both the "RS" and "SR" versions.
- SWEREF93 to RT90/RH70 in Sweden using 20 data points (C.4). The RMS residual distance was 0.169407m for both the "RS" and "SR" versions. This compares with 0.173822m obtained by Andrei. (The latter figure was computed from the rounded RMS components in Table 4.5 of Andrei [2006].)

These results provide strong empirical evidence that the equivalent-enlargement hypothesis is a property of the optimal 8-parameter affine transformation and is therefore a valid constraint in their derivation.

5.4 Derivation of 9-parameter affine transformation

The 9-parameter affine model in Section 2.11 uses a multiplicative matrix of the form **RS** or **SR**. Either way, it is non-linear with respect to the rotation and scaling parameters. This means the least-squares methodology described in Chapter 4 is too simplistic to optimise the parameters.

The methods encountered during this study are listed below. Where *N* is given as the number of control points, it should be read as an arbitrary number not less than 3.

- Späth (2004) describes a method of obtaining the SR version from a dataset of *N* control points. It requires initial approximations of the rotations, but nothing is said about how they are obtained. The process is iterative and – on the evidence of Späth's numerical examples – requires anything from 42 to 137 iterations.
- Andrei (2006) describes a method of obtaining the RS version from a dataset of *N* control points. It requires initial approximations of the scale and rotations, but Andrei says nothing about how they are obtained. Corrections are obtained by least-squares from a linearisation of the observation equations. It is an iterative process and the algorithm (ibid, Appendix B) uses MATLAB software.
- Fan (2009) describes a method of obtaining the RS version from 3 control points. It does not require linearisation or approximate values for the unknown parameters.
- Paláncz *et al* (2008) describes a method of obtaining the SR version from 3 control points. It employs explicit analytical expressions developed by the computer algebra technique "Dixon resultant" as well as by "reduced Groebner basis" for solving the 3-points problem.
- Awange *et al* (2008) gives a "Procrustean solution" for *N* control points. It is sometimes referred to as the ABC-Procrustes algorithm after the names of the authors (Awange, Bae, Claessens). Han (2010) comments that the ABC algorithm "requires iterative computations and only works well in the cases of mild anisotropy". The notation is unusual, but the rotation matrix is applied to coordinates first, making the model equivalent to the SR version.
- Paláncz *et al* (2010) extends the ABC-Procrustes algorithm by the PZ method named after Paláncz and Zaletnyik. It derives the parameters of the SR version from *N* control points. Its improvement on a result in Awange *et al* (2008) indicates that the ABC method does not produce the least-squares solution.

The new method of deriving the optimal 9-parameter transformation has four stages:

- Helmert optimisation: the method of Section 5.1 is applied to obtain the best-fitting 7PC (Helmert) model. Of particular importance are the minimum residual distance and the scale change ΔS_{DA} obtained from distance analysis.
- 8-From-7: an 8-parameter affine transformation is obtained by variations from the best-fitting Helmert model, treating two of the scale factors as equal.

- 9-From-8: a 9-parameter affine transformation is obtained by variations from the 8-parameter affine model (with the individual scale factor derived from the previous stage held fixed).
- 9a-From-9: a further 9-parameter affine transformation is obtained by fixing the ratio of the two scale factors just separated and varying all three scale factors.

For this analysis, the following temporary notation is introduced. d_X, d_Y and d_Z are X, Y and Z components of distance from a data point to the central point in the source datum. d_{XY}, d_{YZ} and d_{XZ} are distance components in the coordinate planes.

The key to the new method is the equivalent-enlargement hypothesis (EEH). This is based on the assumption that the 9-parameter affine transformation that best fits the coordinate data will have the same element of enlargement (or shrinking) as the best-fitting 7PC model.

$$\sum(S_X d_X)^2 + \sum(S_Y d_Y)^2 + \sum(S_Z d_Z)^2 = \sum\left(S_{\mathrm{DA}}\sqrt{d_X^2 + d_Y^2 + d_Z^2}\right)^2. \tag{5-38}$$

An equivalent expression of the equivalent-enlargement hypothesis is

$$S_X^2\sum d_X^2 + S_Y^2\sum d_Y^2 + S_Z^2\sum d_Z^2 = S_{\mathrm{DA}}^2\sum(d_X^2 + d_Y^2 + d_Z^2), \tag{5-39}$$

and it should be noted that the right-hand side of this equation is constant.

The 8-From-7 stage seeks to optimise (at least provisionally) one of the scale factors S_X, S_Y and S_Z of the 9-parameter transformation. The best choice is probably the one corresponding to the smallest spread of values, likely to be the axis-direction which comes closest to being vertical. In the case of the dataset in sub-appendix C.1, the one used in Paláncz *et al* (2010), that is the Y-direction. Without loss of generality, the following description of the 8-From-7 stage treats S_Y as the first scale factor to be optimised. The other scale factor will apply to distances in the XZ plane and will be denoted by S_{XZ}.

Condition (5-39) can be written as

$$S_Y^2\sum d_Y^2 + S_{XZ}^2\sum d_{XZ}^2 = S_{\mathrm{DA}}^2\sum(d_Y^2 + d_{XZ}^2), \tag{5-40}$$

where

$$\sum d_{XZ}^2 = \sum(d_X^2 + d_Z^2) = \sum d_X^2 + \sum d_Z^2. \tag{5-41}$$

Condition (5-41) enables S_{XZ} to be expressed in terms of S_Y:

$$S_{XZ} = \sqrt{\frac{S_{\mathrm{DA}}^2\sum(d_Y^2+d_{XZ}^2) - S_Y^2\sum d_Y^2}{\sum d_{XZ}^2}}, \tag{5-42}$$

and hence ΔS_{XZ} can be expressed in terms of ΔS_Y.

For the moment, MinRMS(ΔS_Y) is defined as the minimised RMS of the distance residual of the 8-parameter affine transformation resulting from a given choice of ΔS_Y. How trial values of ΔS_Y and the resulting ΔS_{XZ} are applied depends on whether the affine model is of type RS (scaling applied first) or SR (rotation applied first).

If the affine model is of type RS, the 6-parameter rigid transformation is treated as that which transforms scaled source coordinates to the target coordinates, where the scaled source coordinates are given by

$$\begin{bmatrix} X_{s,i,sc} \\ Y_{s,i,sc} \\ Z_{s,i,sc} \end{bmatrix} = \begin{bmatrix} S_{XZ} & 0 & 0 \\ 0 & S_Y & 0 \\ 0 & 0 & S_{XZ} \end{bmatrix} \begin{bmatrix} X_{s,i} \\ Y_{s,i} \\ Z_{s,i} \end{bmatrix}. \tag{5-43}$$

If the affine model is of type SR, the 6-parameter rigid transformation is treated as that which transforms the source coordinates to the "unscaled" target coordinates, where the unscaled target coordinates are given by

$$\begin{bmatrix} X_{t,i,unsc} \\ Y_{t,i,unsc} \\ Z_{t,i,unsc} \end{bmatrix} = \begin{bmatrix} 1/S_{XZ} & 0 & 0 \\ 0 & 1/S_Y & 0 \\ 0 & 0 & 1/S_{XZ} \end{bmatrix} \begin{bmatrix} X_{t,i} \\ Y_{t,i} \\ Z_{t,i} \end{bmatrix}. \tag{5-44}$$

Either way, the rigid transformation is optimised by applying stages HO2 & HO3 (described in Section 5.1). For this study, an Excel VBA subroutine was written to perform those stages; it is given in sub-appendix G.2.4.

Note that in the SR case, the translation parameters ΔX, ΔY and ΔZ from the rigid transformation need to be scaled by S_{XZ}, S_Y and S_{XZ} respectively to become part of the 8-parameter set for the affine model.

The obvious starting value of ΔS_Y is ΔS_{DA}, and MinRMS(ΔS_{DA}) is known from the Helmert optimisation. If MinRMS(ΔS_{DA}+0.1ppm) < MinRMS(ΔS_{DA}) then the appropriate search direction will be values of ΔS_Y increasing from ΔS_{DA}. If not, then the appropriate search direction will be values of ΔS_Y decreasing from ΔS_{DA}.

This stage is concluded when the value of ΔS_Y that minimises MinRMS is computed. For the next stage, S_Y is regarded as fixed. The corresponding value of S_{XZ} is assigned to the intermediate quantity S_{int}.

The 9-From-8 stage can be regarded as the optimisation of ΔS_X and ΔS_Z subject to condition (5-39) and S_Y remaining fixed.

Condition (5-39) implies that

$$S_X^2\sum d_X^2 + S_Y^2\sum d_Y^2 + S_Z^2\sum d_Z^2 = S_Y^2\sum d_Y^2 + S_{int}^2\sum(d_X^2 + d_Z^2). \tag{5-45}$$

and hence

$$S_X^2\sum d_X^2 + S_Z^2\sum d_Z^2 = S_{int}^2\sum(d_X^2 + d_Z^2). \tag{5-46}$$

Condition (5-46) enables S_Z to be expressed in terms of S_X:

$$S_Z = \sqrt{\frac{S_{int}^2\sum(d_X^2+d_Z^2) - S_X^2\sum d_X^2}{\sum d_Z^2}}, \tag{5-47}$$

and hence ΔS_Z can be expressed in terms of ΔS_X.

For this stage, MinRMS(ΔS_X) is now re-defined as the minimised RMS of the distance residual of the 9-parameter affine transformation resulting from a given choice of ΔS_X. How trial values of ΔS_X and the resulting ΔS_Z are applied depends on whether the affine model is of type RS (scaling applied first) or SR (rotation applied first).

If the affine model is of type RS, the 6-parameter rigid transformation is treated as that which transforms scaled source coordinates to the target coordinates, where the scaled source coordinates are given by

$$\begin{bmatrix} X_{s,i,sc} \\ Y_{s,i,sc} \\ Z_{s,i,sc} \end{bmatrix} = \begin{bmatrix} S_X & 0 & 0 \\ 0 & S_Y & 0 \\ 0 & 0 & S_Z \end{bmatrix} \begin{bmatrix} X_{s,i} \\ Y_{s,i} \\ Z_{s,i} \end{bmatrix}. \tag{5-48}$$

If the affine model is of type SR, the 6-parameter rigid transformation is treated as that which transforms the source coordinates to the "unscaled" target coordinates, where the unscaled target coordinates are given by

$$\begin{bmatrix} X_{t,i,unsc} \\ Y_{t,i,unsc} \\ Z_{t,i,unsc} \end{bmatrix} = \begin{bmatrix} 1/S_X & 0 & 0 \\ 0 & 1/S_Y & 0 \\ 0 & 0 & 1/S_Z \end{bmatrix} \begin{bmatrix} X_{t,i} \\ Y_{t,i} \\ Z_{t,i} \end{bmatrix}. \tag{5-49}$$

Either way, the rigid transformation is optimised by applying stages HO2 & HO3 (described in Section 5.1).

Note that in the SR case, the translation parameters ΔX, ΔY and ΔZ from the rigid transformation need to be scaled by S_X, S_Y and S_Z respectively to become part of the 9-parameter set for the affine model.

The obvious starting value of ΔS_X is ΔS_{int}, and MinRMS(ΔS_{int}) is the RMS distance residual already computed for the optimised 8-parameter model. If MinRMS(ΔS_{int}+0.1ppm) < MinRMS(ΔS_{int}) then the appropriate search direction will be values of ΔS_X increasing from ΔS_{int}. If not, then the appropriate search direction will be values of ΔS_X decreasing from ΔS_{int}.

This stage is concluded when the value of ΔS_X that minimises MinRMS is computed. The value of S_X, the corresponding value of S_Z and the already-determined value of S_Y will complete the set of scale factors.

The 9a-From-9 stage can be regarded as the optimisation of ΔS_X, ΔS_Y and ΔS_Z subject to condition (5-39) and the ratio S_X: S_Z remaining fixed.

$$c = S_Z/S_X \text{ from the conclusion of the 9-From-8 stage.} \tag{5-50}$$

Keeping the ratio S_X: S_Z fixed becomes the condition

$$S_Z = cS_X. \tag{5-51}$$

Varying S_X requires S_Z to be reset by (5-51) and S_Y to be reset by

$$S_Y = \sqrt{\frac{s_{\mathrm{DA}}^2\Sigma(d_X^2+d_Y^2+d_Z^2)-s_X^2\Sigma d_X^2-s_Z^2\Sigma d_Z^2}{\Sigma d_Y^2}}. \tag{5-52}$$

The obvious starting value of ΔS_X is the value obtained from the 9-From-8 stage. As before, the value is varied until the new MinRMS(ΔS_X) is minimised.

The four-stage method was applied to find the optimal "SR" 9-parameter affine transformation for each of the following datasets (cross-referenced to Appendix C). In each case the parameters obtained were successfully tested by computing the RMS residual distance for scale factors around those obtained by the algorithm. (Six decimal places are shown as in Section 5.3, this time t to be consistent with the sources cited.)

- AGD84 to GDA94 in Western Australia using 82 data points (C.1). The RMS residual distance was 0.733450m. This is less than the RMS residual distances from the

parameters published in Paláncz *et al* (2010) for the same data: 0.733464m (PZ method), 0.733466m (GM method) and 0.749824m (ABC method).

- OSGB36 to WGS84 in Great Britain using 44 data points (C.2). The RMS residual distance was 2.396004m.
- Accra to WGS84 in Ghana's Golden Triangle using 19 data points (C.3). The RMS residual distance was 0.959799m.
- SWEREF93 to RT90/RH70 in Sweden using 20 data points (C.4). The RMS residual distance was 0.178611m. This is identical to the RMS residual distance corresponding to the parameters in Andrei (2006). The values of the parameters differ slightly from those of this study, possibly because Andrei used the "RS" version of the 9-parameter affine transformation.

These results provide strong empirical evidence that the equivalent-enlargement hypothesis is a property of the optimal 9-parameter affine transformation and is therefore a valid constraint in their derivation.

5.5 Derivation of fully-normalised multiple regression equations.

MREs are linear with respect to the parameters and the models are easy to optimise for a given set of basis functions. However, if there are N possible basis functions (and N is potentially infinite), the number of possible combinations is 2^N-1. Accuracy needs to be balanced against other considerations, particularly the desirability of finding a combination with as few terms as possible.

The standard method adopted for this study has been to start with a particular type of model with a specified value of the top power. The parameters are determined by least-squares optimisation based on the control-point data. After each optimisation the most statistically insignificant term is discarded except when

- several terms fail the significance test to such an extent that they can all be discarded (which is most likely to occur after the first optimisation);
- the most statistically-insignificant is sufficiently significant to be retained (meaning no further least-squares optimisation).

The statistical-significance test applied in this study is ratio-less-than-one. The ratio in question is the absolute value of the parameter (which is the coefficient of one of the basis functions) divided by its standard error. For convenience, this can be called the *AP/SE* ratio

or simply *AP/SE*. The method as a whole is best described as *Eliminating Ratios Less Than One* (ERLTO).

If a term has AP/SE less than 1, then its removal should not affect the quality of the model. This is supported by empirical results showing that the next least-squares optimisation will produce a smaller standard error of an observation of unit weight. They also show that discarding a term with AP/SE greater than 1 will increase the standard error of an observation of unit weight.

Discarding all terms with AP/SE less than 1 is best done in stages, because a term whose AP/SE is slightly less than 1 might have a higher AP/SE after less significant terms are discarded. Conversely, a term with AP/SE previously more than 1 might see that ratio drop below 1 after other terms are discarded.

During this study, the criteria adopted for several terms failing the significance test "to such an extent that they can all be discarded" was that AP/SE < 0.2. With the benefit of hindsight, it is acknowledged that a higher threshold (*eg* 0.3) could have been used.

The ERLTO method differs from the process described in Section 4.13 in that no terms are added to the initial set of basis functions. This avoids the problem of deciding which terms are worth adding which is inevitably a trial-and-error process. However, the initial set of basis functions must be smaller than the number of control points, to enable least-squares optimisation. The MREs derived by NIMA/DMA were often based on small quantities of data, so they *had* to start with relatively few basis functions and keep open the possibility of trying new ones after existing terms were discarded.

The ERLTO method breaks down if the standard error of an observation of unit weight is larger than the required accuracy. In that instance a model with more basis functions needs to be introduced. The obvious way of doing this is to raise the top power by 1, assuming that the new set of basis functions will be smaller than the number of control points. If not, a subset of the extra terms could be added, perhaps with priority given to U^iV^j terms for which $i+j$ is smallest.

The end-result of an MRE obtained by ERLTO may be a more accurate model than is actually required. This can be ascertained by one of the quality criteria recommended in Chapter 4,

ideally by the RMS of the residual shifts of the test points (if there are test points extracted from the original dataset). The process of discarding the term with the lowest AP/SE can be resumed to make the model more concise. A suitable title for this exercise is *Additional Trimming*. Sometimes several terms for which AP/SE>1 can be removed before the measure of accuracy is significantly affected.

Additional Trimming can assist comparisons between models of different type (*eg* Ordinary and North/South). If model A gives a lower RMS than model B without a greater number of terms, then model A is superior in terms of accuracy and conciseness. If model A has fewer terms than model B without a higher RMS, then again model A is superior in terms of accuracy and conciseness. However, if model A has a lower RMS and more terms than model B, then comparison requires Additional Trimming of model A until either

- model A has an RMS matching that of model B (leaving the latter as the more concise model) or
- model A has the same number of terms as model B (leaving the former as the more accurate model).

If there are two or more models with more than acceptable accuracy (*ie* RMS lower than required), then Additional Trimming could be applied to each until the RMS reaches the highest acceptable level. The trimmed model with the fewest terms would be the obvious one to select.

Closeness-of-fit and conciseness are not the only considerations when deriving an MRE for modelling a datum shift. Other desirable features are plausible contours and a limited extrapolation effect.

Characteristics of plausible contours are:

- similar shapes, allowing for gradual change across a region;
- directions that change by a relatively small number of degrees (20 degrees or less);
- relatively similar spacing between neighbouring contours.

A limited extrapolation effect means that the highest and lowest coordinate shifts obtained from the model over the intended area of application should not be too far outside the range of shifts for the data points in that area. The largest deviations from that range can be computed from the model at extremity points; the contour pattern and the shape of the area

will inform the selection of “extremity points”. The extremity points can be identified by comparing the outermost contours with the coastal points furthest from them. Allowance should be made for the possibility of more than one extremity point since distance from a contour can be a misleading indication of value.

Sometimes a transformation is known or has been approximated at a set of data points which is sufficiently dense to enable interpolation elsewhere in the area of interest. It is possible that the datum shifts at some or all of the data points have been computed by a method more advanced than those considered in Chapter 2. This chapter considers possible methods of interpolating the datum shifts.

In Chapter 7, some of these methods will be revisited as a means of interpolating *residual* datum shifts at data points, meaning the datum shifts that remain after the application of a transformation model.

6.1 Piecewise 2D affine transformations

As was noted in 2.2, the 2-dimensional affine model given in (2-4) can be used to obtain the transformation exactly from 3 common points, provided they form a triangle (meaning that they are not collinear). From (2-5), it is easily shown that the parameters Δx, Δy, a_1, a_2, a_3 and a_4 are the solution of the following pair of matrix equations:

$$\left.\begin{aligned}\begin{bmatrix}x_{t,1}\\x_{t,2}\\x_{t,3}\end{bmatrix} &= \begin{bmatrix}1 & x_{s,1} & y_{s,1}\\1 & x_{s,2} & y_{s,2}\\1 & x_{s,3} & y_{s,3}\end{bmatrix}\begin{bmatrix}\Delta x\\a_1\\a_2\end{bmatrix}\\ \begin{bmatrix}y_{t,1}\\y_{t,2}\\y_{t,3}\end{bmatrix} &= \begin{bmatrix}1 & x_{s,1} & y_{s,1}\\1 & x_{s,2} & y_{s,2}\\1 & x_{s,3} & y_{s,3}\end{bmatrix}\begin{bmatrix}\Delta y\\a_3\\a_4\end{bmatrix}\end{aligned}\right\} \tag{6-1}$$

The model can then be applied to points on or within the triangle and is, in effect, an interpolation tool.

An important property of the transformation in (2-4) can be deduced by applying it to a point on the side joining vertices 1 and 2 of the triangle in the source datum.

$$\begin{aligned}&\begin{bmatrix}\Delta X\\\Delta Y\end{bmatrix} + \begin{bmatrix}a_1 & a_2\\a_3 & a_4\end{bmatrix}\begin{bmatrix}px_{s,1} + (1-p)x_{s,2}\\py_{s,1} + (1-p)y_{s,2}\end{bmatrix}\\ &\quad= \begin{bmatrix}\Delta X\\\Delta Y\end{bmatrix} + \begin{bmatrix}a_1 & a_2\\a_3 & a_4\end{bmatrix}\begin{bmatrix}px_{s,1}\\py_{s,1}\end{bmatrix} + \begin{bmatrix}a_1 & a_2\\a_3 & a_4\end{bmatrix}\begin{bmatrix}(1-p)x_{s,2}\\(1-p)y_{s,2}\end{bmatrix}\\ &\quad= p\left(\begin{bmatrix}\Delta X\\\Delta Y\end{bmatrix} + \begin{bmatrix}a_1 & a_2\\a_3 & a_4\end{bmatrix}\begin{bmatrix}x_{s,1}\\y_{s,1}\end{bmatrix}\right) + (1-p)\left(\begin{bmatrix}\Delta X\\\Delta Y\end{bmatrix} + \begin{bmatrix}a_1 & a_2\\a_3 & a_4\end{bmatrix}\begin{bmatrix}x_{s,2}\\y_{s,2}\end{bmatrix}\right)\\ &\quad= p\begin{bmatrix}x_{t,1}\\y_{t,1}\end{bmatrix} + (1-p)\begin{bmatrix}x_{t,2}\\y_{t,2}\end{bmatrix}.\end{aligned} \tag{6-2}$$

This means the side joining two vertices in the source datum is transformed linearly to the side joining the transformed vertices in the target datum. So if there are adjoining triangles, as in Figure 6-1, each with its own affine transformation derived from known datum shifts at the vertices, the datum shifts are uniquely defined by linear interpolation along each common side.

It follows that if a set of common points is triangulated and a 2-dimensional affine model is assigned to each triangle, the overall transformation is well-defined and continuous.

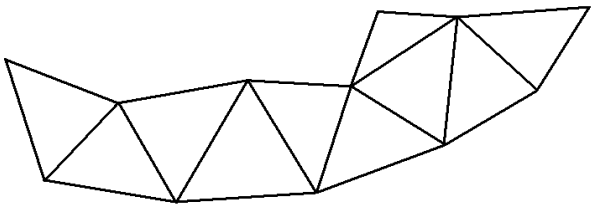


Figure 6-1: Triangles on a plane or surface, each with its own affine transformation derived from known shifts at the vertices.

The problems with this approach include the following:

- The transformation is not smooth because the partial derivatives of the datum shifts are not continuous across the sides of adjoining triangles. Edges (or gradient discontinuities) occur only along lines connecting control points and for no other reason than their connection of control points. This makes the interpolating surface less than the most probable fit.
- Identifying the triangle that contains a particular point is not as straightforward as identifying the rectangle containing the point in a rectangular mesh.

In addition, it should be noted that the transformation is dependent on how the common points are triangulated. In Figure 6-2, the point P could be treated as belonging to triangle ABD or as belonging to triangle ACD. The interpolated datum shifts at point P will differ accordingly.

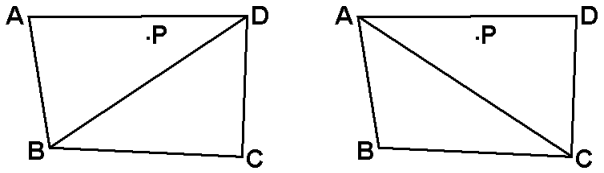


Figure 6-2: Example of a point affected by two different triangulations.

Section 2.2 gave the affine model in terms of grid coordinates. That is precisely what is required if the transformation is grid-to-grid. However, if it is intended for geodetic coordinates (ϕ, λ), there is no need to use a projected grid defined on a literal plane. In (2-4), x can be replaced by ϕ and y by λ (or the other way round, according to personal preference).

6.2 Piecewise bilinear interpolation

Sometimes a transformation has been obtained (or approximated) at the nodes of a regularly-spaced mesh. An example of such a mesh is shown in Figure 6-3.

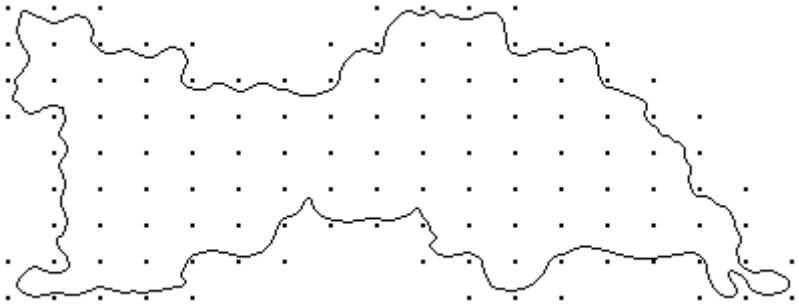


Figure 6-3: Example of a regularly-spaced mesh imposed over an area.

When a function f is known at the vertices of a rectangle ABCD, as shown in Figure 6-4, the bilinear interpolation formula for any point on or within the rectangle is given by

$$f_P = (1-p)(1-q)f_A + p(1-q)f_B + pqf_C + (1-p)qf_D. \quad (6\text{-}3)$$

This is equivalent to linear interpolation along AD and BC followed by linear interpolation along the line through P parallel to AB. It is also equivalent to linear interpolation along AB and DC followed by linear interpolation along the line through P parallel to BC.

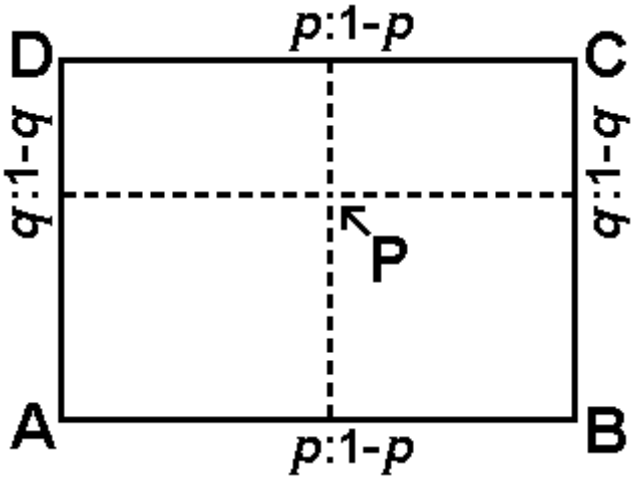


Figure 6-4: Rectangle ABCD with point P between opposite sides with distance ratios p:1-p and q:1-q.

It is easily shown that the interpolation is linear along the sides of the rectangle. If P lies on AB then q=0 and (6-3) simplifies to

$$f_P = (1-p)f_A + pf_B. \quad (6\text{-}4)$$

Similarly, if P lies on BC then p=1 and (6-3) simplifies to

$$f_P = (1-q)f_B + qf_C. \quad (6\text{-}5)$$

Similarly, if P lies on CD then q=1 and (6-3) simplifies to

$$f_P = pf_C + (1-p)f_D. \quad (6\text{-}6)$$

Similarly, if P lies on AD then p=0 and (6-3) simplifies to

$$f_{\mathrm{P}} = (1-q)f_{\mathrm{A}} + qf_{\mathrm{D}}. \qquad (6\text{-}7)$$

It follows that if a set of common points forms a rectangular mesh with bilinear interpolation of the datum shifts Δx and Δy in each rectangle, the overall transformation is well-defined and continuous.

As with piecewise 2D affine transformations, the interpolation is not smooth across a side common to adjoining patches. Unlike the case of partition-by-triangles, however, the rectangle containing any given point is easily identified from the values of the coordinates.

Bilinear interpolation on regularly-spaced control points was used to compute datum shifts in Croatia (Grgić *et al*, 2016). The data had been created from irregularly-spaced data by various composite methods which will be described in Chapter 7.

As acknowledged by Iliffe and Lott (2008, page106), piecewise bilinear interpolation can be applied to geodetic coordinates. The algebraic nature of (6-4) and (6-5) means that $\Delta\phi$ and $\Delta\lambda$ can be interpolated as if the constant-latitude lines were straight and the constant-longitude lines were parallel. (That is how they are sometimes represented on maps.)

6.3 Piecewise bicubic interpolation

This is a more advanced interpolation method when a transformation has been obtained (or approximated) at the nodes of a rectangular mesh.

Piecewise cubic functions are cubic polynomials defined on a series of intervals which join each other at the end-points. They fall into 3 categories:

- Piecewise cubic functions with C^0 continuity; they do no more than join each other and their derivatives are not, in general, continuous.
- Smooth or C^1 piecewise cubic functions; they join each other and have continuous first derivatives.
- Cubic spline functions, also known as cubic splines; these have C^2 continuity which means continuous first and second derivatives.

Piecewise bicubic functions are 2-dimensional cubic polynomials defined on a series of rectangles which join each other along adjacent sides. They fall into 3 categories:

- Piecewise bicubic functions with C^0 continuity; they do no more than join each other and their partial derivatives are not, in general, continuous.
- Smooth or C^1 piecewise bicubic functions; they join each other and have continuous first-order partial derivatives.
- Bicubic spline functions, also known as bicubic splines; these have C^2 continuity which means continuous first-order and second-order partial derivatives.

Piecewise bicubic functions without continuous partial derivatives across the joins have little value in interpolation of datum shifts. At the other extreme, bicubic splines have greater continuity properties than are actually needed; obtaining them involves the solution of large systems of equations.

Consequently, smooth or C^1 piecewise bicubic functions appear to be the most suitable for interpolating datum shifts over a rectangular mesh. There are a number of ways of constructing them. The approach devised for this study is described below. It uses up to 12 surrounding points rather than the 16 envisaged by Iliffe and Lott (2008, page 106).

At the heart of cubic interpolation is the fact that if a function and its derivative are known at both ends of an interval there is one and only one cubic polynomial with the same values and derivatives at those points. Taking the interval as $[x_i, x_{i+1}]$ and writing the polynomial as $\alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3$, the coefficients can be calculated as follows.

$$\alpha_3 = \frac{2\{f(x_i) - f(x_{i+1})\}}{(x_{i+1} - x_i)^3} + \frac{f'(x_i) + f'(x_{i+1})}{(x_{i+1} - x_i)^2} \tag{6-8}$$

$$\alpha_2 = \frac{f'(x_{i+1}) - f'(x_i)}{2(x_{i+1} - x_i)} - \frac{3\alpha_3}{2}(x_i + x_{i+1}) \tag{6-9}$$

$$\alpha_1 = f'(x_{i+1}) - 2\alpha_2 x_{i+1} - 3\alpha_3 x_{i+1}^2 \tag{6-10}$$

$$\alpha_0 = f(x_{i+1}) - \alpha_1 x_{i+1} - \alpha_2 x_{i+1}^2 - \alpha_3 x_{i+1}^3 \tag{6-11}$$

One thing worth noting is the circumstance that makes the interpolating function a quadratic polynomial. From (6-8),

$$\alpha_3 = 0 \text{ if and only if } f'(x_i) + f'(x_{i+1}) = \frac{2\{f(x_{i+1}) - f(x_i)\}}{(x_{i+1} - x_i)}. \tag{6-12}$$

A simple way of assigning derivatives at equally-spaced points is shown in Figure 6-5. A symmetric formula based on 2 points equidistant from x_i is used at all points except the first and last. The formulae used at the end points will ensure that the cubic polynomials in the end intervals are quadratic.

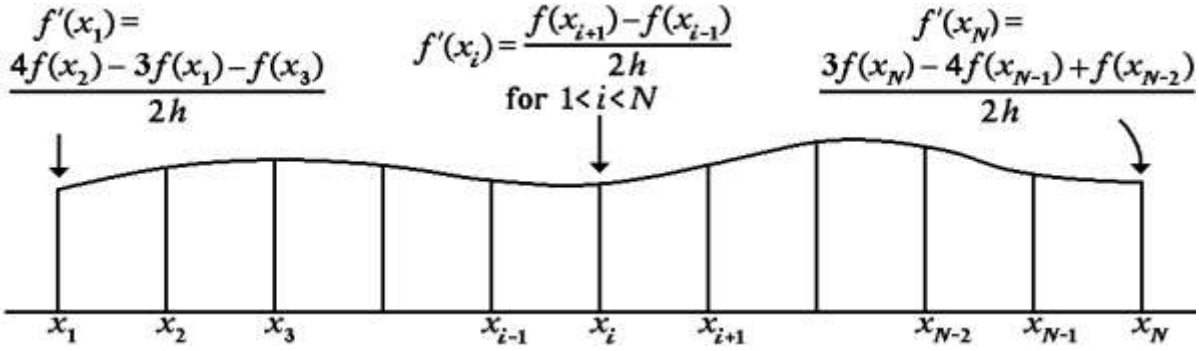


Figure 6-5: Simple derivative approximations over a set of equally-spaced points with an interval h.

Whether the derivatives are assigned as in Figure 6-5 or by some other process, the cubic polynomials fitted to the respective intervals will be smooth at the points where f is defined.

This method of cubic interpolation can be extended to bicubic interpolation covering a rectangle.

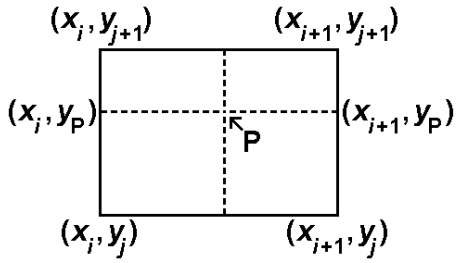


Figure 6-6: Point P in the sub-rectangle of a mesh with bottom-left vertex (x_i, y_j).

Consider a point P that falls within a rectangle for which f, $\partial f/\partial x$ and $\partial f/\partial y$ are known at the four vertices, as illustrated in Figure 6-6. The interpolated value of f at P is obtained as follows.

- Using the values of f and $\partial f/\partial y$ at the left-hand vertices, an interpolating cubic polynomial for f is computed and evaluated at (x_i, y_P).
- Using the values of f and $\partial f/\partial y$ at the right-hand vertices, an interpolating cubic polynomial for f is computed and evaluated at (x_{i+1}, y_P).
- Using the values of $\partial f/\partial x$ at the left-hand vertices, a linearly interpolated value of $\partial f/\partial x$ is evaluated at (x_i, y_P).
- Using the values of $\partial f/\partial x$ at the right-hand vertices, a linearly interpolated value of $\partial f/\partial x$ is evaluated at (x_{i+1}, y_P).
- Using the interpolated values of $\partial f/\partial x$ and $\partial f/\partial y$, an interpolating cubic polynomial for f is computed and evaluated at P.

The simplest method of assigning partial derivatives is to use symmetric two-point approximations where possible and one-sided approximations where not possible. It is suitable for meshes like the one shown in Figure 6-3 as well as those with a rectangular boundary.

The result is that for any rectangle, a maximum of 12 values of *f* contribute to an interpolated value in that rectangle, as illustrated by the 12 vertices in Figure 6-7. The 12 values come from those 12 vertices.

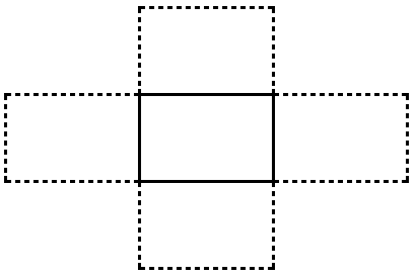


Figure 6-7: A rectangle with its fully-adjacent neighbours.

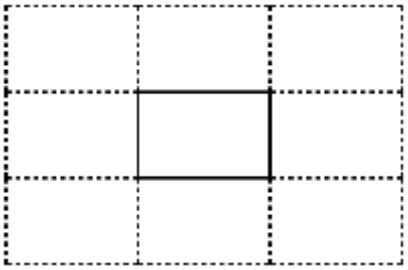


Figure 6-8: A rectangle with its fully- and diagonally-adjacent neighbours.

Some mathematicians prefer bicubic interpolation that makes use of up to 16 values of *f*, as illustrated by the 16 vertices in Figure 6-8. The disadvantage is that a more detailed algorithm is needed to cover the cases where less than 16 values are available, because they will be more numerous than the cases involving less than 12. This is particularly true when the mesh does not have a rectangle as its boundary, as is the case in Figure 6-3.

Piecewise bicubic interpolation can be applied to geodetic coordinates. $\Delta\phi$ and $\Delta\lambda$ can be interpolated as if the constant-latitude lines were straight and the constant-longitude lines were parallel. Those lines only have those properties in a graphical representation, but that is good enough for cubic interpolation.

6.3.1 Mesh-of-partial-derivatives implementation

One implementation is to start with a one-off computation of the partial derivatives at every mesh point, so that every rectangle is ready for interpolation. $\partial f/\partial x$ and $\partial f/\partial y$ are computed numerically by formulae (6-13) to (6-22) at each point (x_i, y_j) where *f* is known. (The reason for not using *h* in the formulae is that the interval between mesh points is not necessarily the same in the *x* and *y* directions.)

If *f* is known at (x_{i-1}, y_j) and (x_{i+1}, y_j),

$$\left(\frac{\partial f}{\partial x}\right)_{i,j} = \frac{f(x_{i+1},y_j)-f(x_{i-1},y_j)}{x_{i+1}-x_{i-1}}; \tag{6-13}$$

otherwise if f is known at (x_{i+1}, y_j) and (x_{i+2}, y_j),

$$\left(\frac{\partial f}{\partial x}\right)_{i,j} = \frac{4f(x_{i+1},y_j)-3f(x_i,y_j)-f(x_{i+2},y_j)}{x_{i+2}-x_i}; \tag{6-14}$$

otherwise if f is known at (x_{i-2}, y_j) and (x_{i-1}, y_j),

$$\left(\frac{\partial f}{\partial x}\right)_{i,j} = \frac{3f(x_i,y_j)-4f(x_{i-1},y_j)+f(x_{i-2},y_j)}{x_i-x_{i-2}}; \tag{6-15}$$

otherwise if f is known at (x_{i+1}, y_j),

$$\left(\frac{\partial f}{\partial x}\right)_{i,j} = \frac{f(x_{i+1},y_j)-f(x_i,y_j)}{x_{i+1}-x_i}; \tag{6-16}$$

otherwise if f is known at (x_{i-1}, y_j).

$$\left(\frac{\partial f}{\partial x}\right)_{i,j} = \frac{f(x_i,y_j)-f(x_{i-1},y_j)}{x_i-x_{i-1}}. \tag{6-17}$$

If f is known at (x_i, y_{j-1}) and (x_i, y_{j+1}),

$$\left(\frac{\partial f}{\partial y}\right)_{i,j} = \frac{f(x_i,y_{j+1})-f(x_i,y_{j-1})}{y_{j+1}-y_{j-1}}; \tag{6-18}$$

otherwise if f is known at (x_i, y_{j+1}) and (x_i, y_{j+2}),

$$\left(\frac{\partial f}{\partial y}\right)_{i,j} = \frac{4f(x_i,y_{j+1})-3f(x_i,y_j)-f(x_i,y_{j+2})}{y_{j+2}-y_j}; \tag{6-19}$$

otherwise if f is known at (x_i, y_{j-2}) and (x_i, y_{j-1}),

$$\left(\frac{\partial f}{\partial y}\right)_{i,j} = \frac{3f(x_i,y_j)-4f(x_i,y_{j-1})+f(x_i,y_{j-2})}{y_j-y_{j-2}}; \tag{6-20}$$

otherwise if f is known at (x_i, y_{j+1}),

$$\left(\frac{\partial f}{\partial y}\right)_{i,j} = \frac{f(x_i,y_{j+1})-f(x_i,y_j)}{y_{j+1}-y_j}; \tag{6-21}$$

otherwise if f is known at (x_i, y_{j-1}),

$$\left(\frac{\partial f}{\partial y}\right)_{i,j} = \frac{f(x_i,y_j)-f(x_i,y_{j-1})}{y_j-y_{j-1}}. \tag{6-22}$$

The result of compiling a complete mesh of both $\partial f/\partial x$ and $\partial f/\partial y$ values is that for any point of interest, the enclosing rectangle will have values of f, $\partial f/\partial x$ and $\partial f/\partial y$ at its vertices. The bicubic interpolation method can be employed at the point of interest.

6.3.2 Partial-derivatives-as-required implementation

The alternative implementation is to restrict computation of the partial derivatives to the vertices of the rectangle containing the latest point of interest. This involves more coding than the mesh-of-partial-derivatives implementation, but less actual computation if the number of points of interest is less than a quarter of the number of mesh points.

The following procedure is applied for each point P(x, y) where an interpolated estimate of f is required:

- Values of i_P and j_P are computed such that $x_{i_\mathrm{P}} \leq x \leq x_{i_\mathrm{P}+1}$ and $y_{j_\mathrm{P}} \leq y \leq y_{j_\mathrm{P}+1}$. This identifies the mesh-point rectangle to which point P belongs, by making i_P and j_P the subscripts of the bottom-left point.
- The partial derivatives at the bottom-left point are obtained by applying the process in formulae (6-13) to (6-22) with $i = i_\mathrm{P}$ and $j = j_\mathrm{P}$.
- The partial derivatives at the bottom-right point are obtained by applying the process in formulae (6-13) to (6-22) with $i = i_\mathrm{P} + 1$ and $j = j_\mathrm{P}$.
- The partial derivatives at the top-left point are obtained by applying the process in formulae (6-13) to (6-22) with $i = i_\mathrm{P}$ and $j = j_\mathrm{P} + 1$.
- The partial derivatives at the top-right point are obtained by applying the process in formulae (6-13) to (6-22) with $i = i_\mathrm{P} + 1$ and $j = j_\mathrm{P} + 1$.

The bicubic interpolation method can then be employed at the latest point of interest, but the partial derivatives are not stored for future use.

6.3.3 Implementation with a surrounding rectangle of grid points

For this option, a prerequisite is that the function values at grid points are derived from a model. That being the case, it is usually possible to generate values over an enlarged grid. The enlargement considered here is a wholly rectangular mesh with a surrounding corridor which is outside all the rectangles where interpolation is intended. This is illustrated in Figure 6-9: the shaded rectangles are the rectangles shown in Figure 6-3.

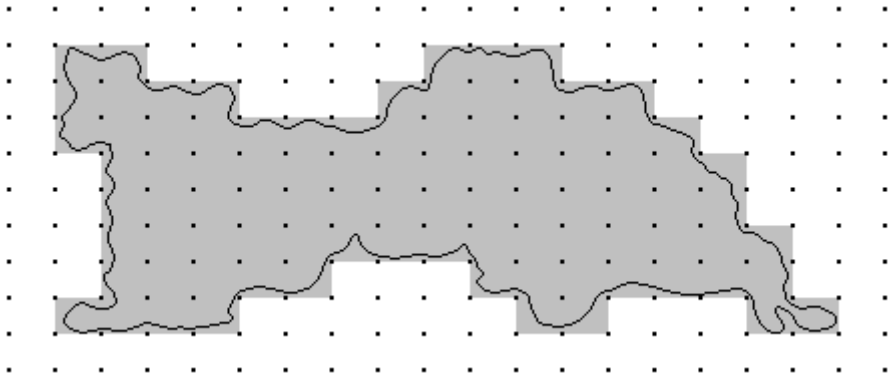


Figure 6-9: Example of a regularly-spaced mesh enlarged to ensure a surrounding rectangular corridor.

The procedure in subsection 6.3.1 simplifies to deriving partial derivatives from (6-13) and (6-18) at all non-perimeter grid points, storing them in arrays, then using them during the interpolation.

The alternative procedure in subsection 6.3.2 simplifies to deriving partial derivatives from (6-13) and (6-18) at the enclosing four grid points for each point where interpolation is required.

It is imperative that any software package based on this approach ensures that interpolation is only performed in the rectangles for which the model was intended. In the case of Figure 6-9, that means the shaded rectangles.

One point that needs to be carefully considered is the quality of the function values at the extra grid points neighbouring the original grid points. In Figure 6-9, this means the function values at the outer vertices of unshaded rectangles just outside the shaded rectangles. If the model is stable, the partial derivatives should compare favourably with those computed from equations (6-14) to (6-17) and (6-19) to (6-22). If not, it might be safer to use either of the more intricate procedures in subsections 6.3.1 and 6.3.2.

CHAPTER 7: COMPOSITE TRANSFORMATION METHODS

With the possible exception of multiple regression equations, the methods discussed in Chapter 2 are not well-suited to modelling transformations with distortions. Even MREs are limited in the extent they can represent distortion, and there is a danger that high-degree polynomials will introduce distortions of their own. To represent datum shifts with distortions accurately, computations need to be based on more advanced methods or a combination of techniques.

The starting point for deriving models is a set of data points where the coordinates have been obtained in both datums. Where all of them are used to predict transformations, the terms "data points" and "control points" are synonymous; this is the case for Western Australia (Chapter 8) where there are only 82 data points. Only for the largest datasets (Chapters 12 and 13) have the data points been split into control points and test points; the latter do not contribute to the surface-fitting and provide an independent check on the accuracy of the methods.

Much of this chapter is devoted to transformation-enhancement methods that fit the residuals from an initial model either exactly by interpolation or approximately by smoothing. Some of the methods could theoretically be used as transformation methods in their own right (*ie* as enhancements of the identity function), but are not considered as such here. Enhancement methods in this chapter are derived from the residuals of a "trend" model.

Grgić *et al* (2016) is notable for a list of distortion-modelling techniques which are applied to the HDKS→HTRS96 datum transformation. They chose a 7PC model, as described in 2.6 above, for a least-squares fit to the datum shifts at common points (although it is possible that the least-squares fit was obtained from the Bursa-Wolf form). The distortions they modelled were residuals from that model after the coordinates had been converted to projected grid coordinates.

The software used by Grgić *et al* was Surfer in all cases except least-squares collocation. Surfer is a contouring and 3D surface mapping program developed by the US Golden Company.

Advanced methods often require distance computations between points where datum shifts are known and points where datum shifts are required. The options for distance computations include the following:

- Distances computed on a projected grid; the actual computation is simple (Pythagoras' formula) but the method does require the choice of a projection, making it less general.
- Distances computed from Cartesian coordinates; again, the computation is simple, but they are lengths of chords rather than lines on a curved surface and the difference is significant over large areas.
- Ellipsoidal distances computed by a geodesic formula, which involves more complicated mathematical expressions that the required accuracy needs.
- Approximate ellipsoidal distances; this is examined in Appendix B. Applying (B-11), (B-10) and (B-14), the best formula for the distance between points *i* and *j* appears to be

$$d_{i,j} = c_{i,j} + \frac{c_{i,j}^3}{24R^2} + \frac{3c_{i,j}^5}{640R^4}, \tag{7-1}$$

where

$$c_{i,j} = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2 + (Z_j - Z_i)^2} \tag{7-2}$$

and

$$R^2 = \frac{a^2(1-e^2)}{(1-e^2\sin^2\phi_i)(1-e^2\sin^2\phi_j)}. \tag{7-3}$$

Distance between points is generally assumed to be a factor in the correlation of datum shifts. That is to say that a datum shift is more likely to be close to the shift at a nearby point than to the shift at a faraway point. Unless there is reason to suppose otherwise, the distance-dependence of correlation is assumed to be independent of direction. In other words, the spatial correlation structure is *isotropic*.

Some methods in Surfer allow users to specify either isotropic correlation (which is independent of direction) or anisotropic correlation (which is dependent on direction). Surfer (2002, page 109) defines anisotropy ratio as "the maximum range divided by the minimum range" and anisotropy angle as "the preferred orientation (direction) of the major axis in degrees". Users can opt for isotropy by assigning the default values 1 to anisotropy ratio and 0 to anisotropy angle. This was done by Grgić *et al* (2016). One of the likely reasons for isotropy is that units in the *x* and *y* directions were the same.

Correlation is often measured by *covariance*, of which *variance* is a limiting value. Central to both concepts is the mathematical expectation operator value denoted by $E[\ldots]$ in Blais (1982) and Cross (1983). Its properties – including linearity – are outlined in Cross (1983, Section 4.1). The expected value of a position-dependent variable is its mean, meaning its average value over its domain.

- Covariance is the mean value of the product of the deviations of two variates from their respective means:

$$Cov(\alpha, \beta) = E[(\alpha - E[\alpha])(\beta - E[\beta])]. \tag{7-4}$$

- Variance is the expectation of the squared deviation of a variate from its mean:

$$Var(\alpha) \equiv Cov(\alpha, \alpha) = E[(\alpha - E[\alpha])^2]. \tag{7-5}$$

Correlation between elements of separate vectors can be represented by a *covariance matrix*. The covariance matrix $\mathbf{Cov}(\mathbf{v},\mathbf{w})$ of two vectors $\mathbf{v}$ and $\mathbf{w}$ is defined to be the matrix of covariances of the pairs that make up the products in $\mathbf{v}\mathbf{w}^T$. If the vectors have m and n elements respectively, then

$$\mathbf{Cov}(\mathbf{v},\mathbf{w}) = \begin{bmatrix} Cov(v_1,w_1) & Cov(v_1,w_2) & \cdots & Cov(v_1,w_n) \\ Cov(v_2,w_1) & Cov(v_2,w_2) & \cdots & Cov(v_2,w_n) \\ \vdots & \vdots & & \vdots \\ Cov(v_m,w_1) & Cov(v_m,w_2) & \cdots & Cov(v_m,w_n) \end{bmatrix}. \tag{7-6}$$

The special case $\mathbf{Cov}(\mathbf{v},\mathbf{v})$ is sometimes referred to as a variance-covariance matrix (VCV matrix), because the diagonal elements are variances.

$$\mathbf{Cov}(\mathbf{v},\mathbf{v}) = \begin{bmatrix} Cov(v_1,v_1) & Cov(v_1,v_2) & \cdots & Cov(v_1,v_n) \\ Cov(v_2,v_1) & Cov(v_2,v_2) & \cdots & Cov(v_2,v_n) \\ \vdots & \vdots & & \vdots \\ Cov(v_n,v_1) & Cov(v_n,v_2) & \cdots & Cov(v_n,v_n) \end{bmatrix}. \tag{7-7}$$

For the most part, the advanced methods are used to generate *gridded* data, in the sense of data based on a regular mesh of points that is suitable for piecewise interpolation. The process of generating such data is called *gridding*, although it is not to be confused with the term *projection* that converts geodetic coordinates to grid coordinates.

The strategy adopted in this study was to perform gridding through interpolation rather than approximation at the control points. In general, the gridded data will not include control points, so piecewise interpolation will generate approximate values at control points. This, however, is approximation derived from a model that can claim to be an exact fit (at least at the control points). If the gridding had been done by an approximate method based on

smoothing, then the subsequent piecewise interpolation of the gridded data would be approximation derived from what is already an approximation.

The use of interpolation rather than smoothing on control points does have its dangers. Some methods are "not strictly bounded", in the sense that they introduce peaks and troughs outside the range of values at control points. If the range is widened only slightly, this may be acceptable; perhaps even desirable; see the extremities between P & Q and between S & T in Figure 7-1. In this case, there is unlikely to be an adverse effect on gridding. If the range is widened more than slightly, then the interpolation method is liable to introduce distortions beyond what is reasonable to deduce from the data. An example of this is shown in Figure 7-2, with extremities between Q & R and between S & T. A volatility analysis may be necessary to test for unacceptable peaks and troughs where the data patterns indicate that they are most likely to occur.

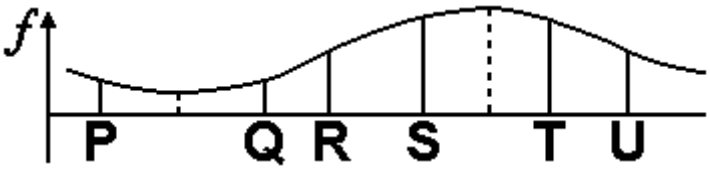


Figure 7-1: Acceptable extremities introduced by interpolation along a row of collinear control points.

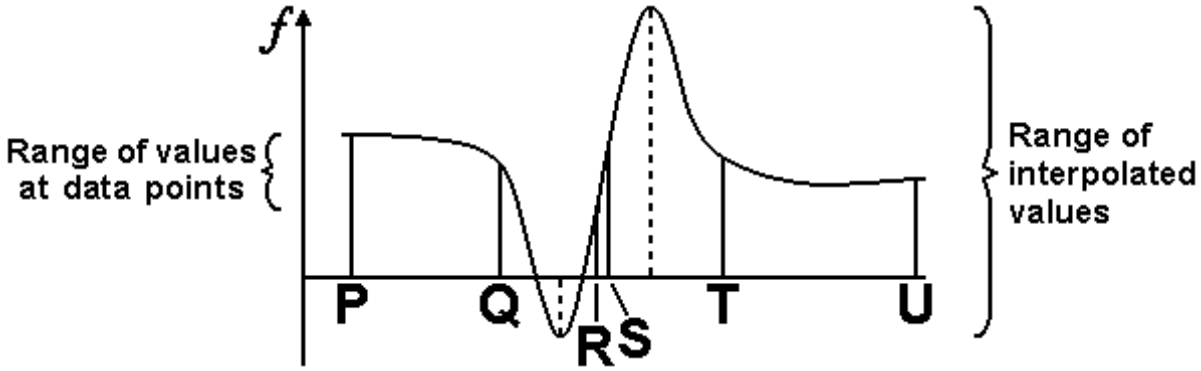


Figure 7-2: Possibly unacceptable extremities introduced by interpolation along a row of collinear control points.

Some methods of interpolation, such as inverse distance to a power and nearest neighbour, are "strictly bounded" and cannot introduce values outside the range of values at control points. In general, they are based on a linear combination of data-point values where the coefficients are non-negative and add up to 1. These methods avoid the kind of oscillations that occur in Figure 7-2. On the other hand, they would not interpolate the data shown in Figure 7-1 as well as the curve shown; the interpolant would either be constant between S and T, or would dip in value between them.

The list of methods described is not exhaustive; there are others that could be investigated, including neural networks. A neural network is a form of artificial intelligence that relies on a group of interconnected mathematical equations that accept input data and calculate an output. Applications where they have a possible use include those where relationships are difficult to describe adequately with conventional approaches.

7.1 Transformations by least-squares collocation

Datum transformations by least-squares collocation are different from most composite methods. There is a similarity with transformations by a "trend" model which are followed by a method to improve on the accuracy from that model. The difference is that least-squares collocation performs the two stages in parallel, unless a model has been removed in advance.

Ruffhead (1987) described the general starting point for least-squares collocation as the equation

$$measurement = mathematical\ model + signal + noise. \tag{7-8}$$

This assumes that the measurements at the data points are assumed to contain unbiased measurement errors ("noise") as well as an unmodelled component (the "signal"). Least-squares collocation estimates the model and the signal at the control points (which – by allowing for noise – amounts to a smoothing of the measurements rather than an interpolation) and predicts the value of the model + signal at a set of computation points. González-Matesanz *et al* (2003, 2006) use least-squares collocation for datum shifts ($\Delta\phi$, $\Delta\lambda$) between ED50 and ETRS89.

The mathematical model is designed to have the same mathematical expectation as the measured quantity. This means that the sum of the signal and noise have zero expectation. Given the bias-free nature of noise, it must have zero expectation. As a result, signal has zero expectation. The effect of this on covariance and variance is as follows:

- Applying (7-4), the covariance of the signals at points *i* and *j* is given by
$$Cov(s_i, s_j) = E[s_i s_j]. \tag{7-9}$$
- Applying (7-5), the variance of the signal at points *i* is given by
$$Var(s_i) \equiv Cov(s_i, s_i) = E[s_i^2]. \tag{7-10}$$

A key element of least-squares collocation is modelling of the signal's covariance by a covariance function. This treats the signal as correlated with distance, so that for each

computation point, the closest control points have the strongest effect. The usefulness of a covariance function depends on how well it represents correlation. The distance for which covariance equals half the variance is called the *correlation length*, and it is important that the covariance function reflects this.

Using a covariance function, signal covariance is the expected value of the product of two signals separated by a given distance. If there is sufficient data, an empirical value can be obtained from the mean of the products of two signals separated by a range of distances whose central value is the given distance.

The variance of the signal is the expected value of the signal squared, the limiting value of the covariance as distance approaches zero. The variance will be denoted C_0 and will generally be included in any covariance function.

A covariance function needs at least one other parameter which determines the shape of the covariance curve plotted against distance. It may be "correlation length" which is the distance at which covariance is half the variance. It may be the distance at which covariance becomes zero, although that option is not feasible if covariance approaches zero asymptotically (as in Figure 7-3). Either of these types can be regarded as a *shape parameter*, which controls how rapidly the function changes: increasing the value of the shape parameter has the effect of stretching the covariance curve.

Example 7.1: Gaussian covariance function.

The simplest covariance functions are infinite in the sense that they approach zero as distance approaches infinity. The most commonly used of these is the Gaussian function

$$C(r) = C_0 \exp(-k^2 r^2). \tag{7-11}$$

The inverse of the Gaussian covariance function – which provides the distance corresponding to a given covariance - can be obtained by rearrangement of (7-11):

$$r = (1/k) \cdot \sqrt{\ln(C_0/C)}. \tag{7-12}$$

The defining constant *k* is related to the correlation length by the fact that their product is $\sqrt{\ln 2}$ or 0.8325546. The function is illustrated in Figure 7-3.

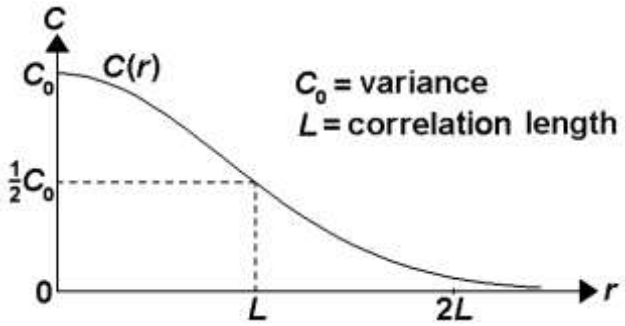


Figure 7-3: Gaussian covariance function.

The rate of decay of the Gaussian covariance function can be gauged from the following fact. If L is the value of r for which $C(r) = 0.5C_0$, then $C(2L) = 0.0625C_0$, $C(3L) = 0.001953C_0$ and $C(4L) = 0.000015C_0$.

The defining constant k in (7-11) can be determined from an empirically-derived covariance C_1 corresponding to a particular distance r_1. Applying (7-12),

$$k = (1/r_1) \cdot \sqrt{\ln(C_0/C_1)}. \tag{7-13}$$

Example 7.2: Finite covariance function based on cubic splines (SS20).

Finite covariance functions are those for which *C(r)* vanishes beyond a certain distance. That distance is described here as "limit-of-influence" and denoted by r_{max}. Finite covariance functions have been suggested by Sansò and Schuh (1987). One example is their Equation (20) which has a smooth join at $r = 0.5r_{max}$. They proposed it for a variable based on a one-dimensional domain. For this study it was investigated over a surface because of its relative simplicity. This covariance function is the right-hand half of a cubic B-spine than can be defined as follows.

$$C(r) = \begin{Bmatrix} C_0[1 - 6r^2(r_{max} - r)/r_{max}^3] & \text{for } 0 \le r \le 0.5r_{max}; \\ 2C_0(1 - r/r_{max})^3 & \text{for } 0.5r_{max} \le r \le r_{max}; \\ 0 & \text{for } r \ge r_{max} \end{Bmatrix} \tag{7-14}$$

For this function, the defining constant is r_{max}. The correlation length is $0.361176r_{max}$ and r_{max} is 2.768733 times the correlation length. The function is illustrated in Figure 7-4.

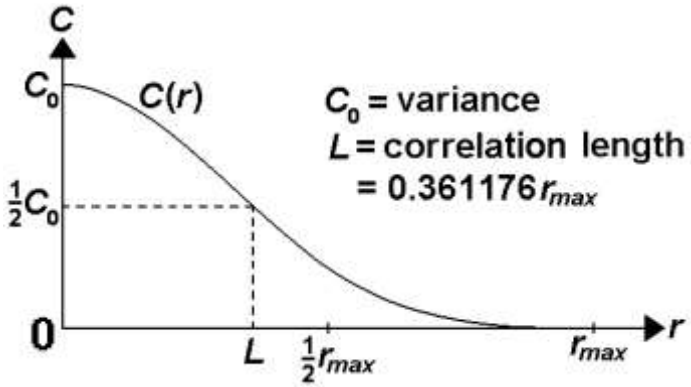


Figure 7-4: Cubic-spline (SS20) covariance function.

The inverse of the SS20 covariance function – which provides the distance corresponding to a given covariance – is as follows:

$$r = \left\{ \begin{array}{cc} r_{max}\{1 - \sqrt[3]{0.5C/C_0}\} & \text{for } 0 \le C \le 0.25C_0; \\ r_{max}\{0.6954 - 0.9563C/C_0 + 0.8213(C/C_0)^2 - 0.4917(C/C_0)^3\} & \text{for } 0.25C_0 \le C \le 0.85C_0; \\ r_{max}\{0.4493\sqrt{1 - C/C_0} - 0.148(C/C_0 - 0.85)[1 - C/C_0]^{0.4}\} & \text{for } 0.85C_0 \le C \le C_0. \end{array} \right\} \quad (7\text{-}15)$$

Only the first expression is exact. The other two expressions are close numerical approximations. They were derived for this study by the following procedure:

- In each of the ranges, inverses were calculated and tabulated by means of Newton-Raphson iteration (using approximate rather than exact derivatives).
- The interpolating function in the central section was constructed from a linear function joining the end-points and a cubic polynomial to approximate the residuals. The polynomial was divisible by $C/C_0 - 0.25$ and $0.85 - C/C_0$, so there were two parameters which could be chosen to give the closest possible fit to the tabulated inverses.
- The interpolating function in the final section was constructed from the $\sqrt{1 - C/C_0}$ term (weighted to ensure it joins the end points) and a weighted correction term to approximate the residuals. The power 0.4 was chosen as a compromise between a cube root and a square root (giving a better fit than either).

The defining constant r_{max} in (7-14) can be determined from an empirically-derived covariance C_1 corresponding to a particular distance r_1. Applying (7-15),

$$r_{max} = \left\{ \begin{array}{c} r_1/\left[1 - \sqrt[3]{C_1/(2C_0)}\right] \quad \text{if } 0 \le C_1 \le 0.25C_0; \\ r_1/[0.6954 - 0.9563C_1/C_0 + 0.8213(C_1/C_0)^2 - 0.4917(C_1/C_0)^3] \\ \quad \text{if } 0.25C_0 \le C_1 \le 0.85C_0; \\ r_1/\left\{0.4493\sqrt{1 - C_1/C_0} - 0.148(C_1/C_0 - 0.85)[1 - C_1/C_0]^{0.4}\right\} \\ \quad \text{if } 0.85C_0 \le C_1 \le C_0. \end{array} \right\} \quad (7\text{-}16)$$

If a single empirical covariance is being used to determine r_{max} (and hence the covariance model) then it is desirable that it lies in the range $0.3C_0$ to $0.7C_0$. That range covers the steepest part of the curve in Figure 7-4, so the effect of any error in the empirical covariance will be relatively small.

Example 7.3: Finite covariance function based on Sansò and Schuh's Equation 30 (SS30).

This too has a limit-of-influence denoted by r_{max}. Equation 30 of Sansò and Schuh (1987) uses a single analytical expression over the range of distances (which it calls "finite support"). Arabelos and Tscherning (1996) use it as a covariance function in the context of gravity modelling. The version considered here is the normalised form given in equation (7-17).

$$C(r) = \begin{cases} C_0[1-6(r/r_{max})^2][1-(2/\pi)\arcsin(r/r_{max})] \\ \quad +C_0[6(r/r_{max})+32(r/r_{max})^3-8(r/r_{max})^5]\sqrt{1-(r/r_{max})^2}/(3\pi) \\ \qquad\qquad\qquad\qquad\qquad\qquad \text{for } 0 \le r \le r_{max}; \\ 0 \qquad\qquad\qquad\qquad\qquad\qquad \text{for } r \ge r_{max}. \end{cases} \tag{7-17}$$

For this function, as given in (7-17), the defining constant is r_{max}. (The constant preferred by Sansò and Schuh is R, where $R = 0.5r_{max}$.) The correlation length is $0.374712r_{max}$ and is r_{max} is 2.668717 times the correlation length. The function is illustrated in Figure 7-5.

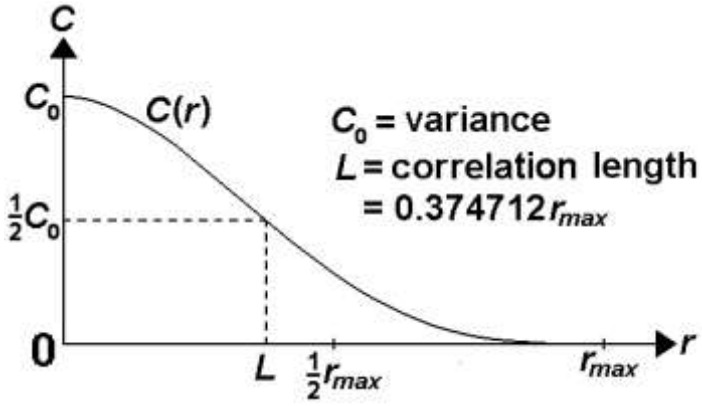


Figure 7-5: SS30 covariance function.

The inverse of the SS30 covariance function – which provides the distance corresponding to a given covariance – is as follows:

$$r = \begin{cases} r_{max}\left[1-\sqrt[3]{0.44128C/C_0-0.31784(C/C_0)^2}\right] & \text{for } 0 \le C \le 0.1C_0; \\ r_{max}(0.71965-0.64325C/C_0) & \\ -r_{max}(C/C_0-0.1)(0.9-C/C_0)\times & \\ [0.7943-2.5076C/C_0+3.4212(C/C_0)^2-2.009(C/C_0)^3] & \text{for } 0.1C_0 \le C \le 0.9C_0; \\ r_{max}\sqrt{0.18622(1-C/C_0)+0.11795(1-C/C_0)^2} & \text{for } 0.9C_0 \le C \le C_0. \end{cases} \tag{7-18}$$

All 3 expressions are close numerical approximations which are exact at the joins. They were derived for this study by the following procedure:

- In each of the ranges, inverses were calculated and tabulated by means of Newton-Raphson iteration (using approximate rather than exact derivatives).
- The interpolating function in the first section was designed to be $1-\sqrt[3]{aC/C_0+b(C/C_0)^2}$ where the square term was deemed to be a correction. The constants *a* and *b* were chosen to ensure that the function was correct for $C=0.1C_0$ and that the residuals were minimised.
- The interpolating function in the central section was made the sum of a linear function joining the end-points and a 5th-order polynomial divisible by $C/C_0-0.1$ and $0.9-C/C_0$. The four outstanding parameters were chosen to give the closest possible fit to the tabulated inverses.
- The interpolating function in the final section was designed to be $\sqrt[3]{a(1-C/C_0)+b(1-C/C_0)^2}$ where the square term was deemed to be a correction. The constants *a* and *b* were chosen to ensure that the function was correct for *C*=0.9 and that the residuals were minimised.

The defining constant r_{max} in (7-17) can be determined from an empirically-derived covariance C_1 corresponding to a particular distance r_1. Applying (7-18),

$$r_{max}=\left\{\begin{array}{ll} r_1/\left[1-\sqrt[3]{0.44128C_1/C_0-0.31784(C_1/C_0)^2}\right] & \text{if } 0\le C\le 0.1C_0; \\ r_1/\{(0.71965-0.64325C_1/C_0)-(C_1/C_0-0.1)(0.9-C_1/C_0)\times & \\ \quad [0.7943-2.5076C_1/C_0+3.4212(C_1/C_0)^2-2.009(C_1/C_0)^3]\} & \\ & \text{if } 0.1C_0\le C\le 0.9C_0; \\ r_1/\sqrt{0.18622(1-C_1/C_0)+0.11795(1-C_1/C_0)^2} & \text{if } 0.9C_0\le C\le 0.1C_0. \end{array}\right\} \quad (7\text{-}19)$$

If a single empirical covariance is being used to determine r_{max} (and hence the covariance model) then it is desirable that it lies in the range $0.3C_0$ to $0.7C_0$. That range covers the steepest part of the curve in Figure 7-5, so the effect of any error in the empirical covariance will be relatively small.

(End of examples)

The correlation length can be estimated by sampling, but in practice it is often guessed from the spread of data points. One method devised for this study is to calculate each data point's distance to the nearest neighbouring data point, then set the correlation length to the median value of those distances. The median distance to nearest neighbour (MDNN) reflects only the spread of data points and not the behaviour of the signal. Nevertheless, it provides a useful initial estimate.

If the measurements at the control points are treated as error-free, equation (7-8) simplifies to

(errorless) measurement = mathematical model + signal. (7-20)

Least-squares collocation estimates the model and the signal at the control points (which in this case are the components of an interpolation) and predicts the value of the model + signal at a set of computation points. Applied to transformations, the entire distortion is treated as a signal, and the signal interpolates residuals from the trend model.

A special case is when the measurements have been "centred". That could have occurred by the prior removal of a trend model obtained by least-squares estimation. In this case, the starting point is

(centred, errorless) measurement = signal. (7-21)

This special case has been used by a number of authors to derive transformations after the removal of a model. They include:

- You and Hwang (2006) for grid coordinate transformation between two geodetic datums of Taiwan; the trend model is 2D conformal as in Section 2.1. This paper considers both the case of signal only and signal + noise.
- Yun *et al* (2006) for distortion modelling in Korea; the trend model is Molodensky-Badekas, described in Section 2.9 and the collocation is applied to latitude and longitude rather than projected grid coordinates.
- Grgić *et al* (2016) for distortion modelling in Croatia, although the trend model was in terms of 3D coordinates while the collocation was applied to grid coordinates. The software used was GRAVSOFT, documented by Forsberg and Tscherning (2008), and in particular the routine GEOGRID. This uses a quadrant-search method to speed up computations: at each computation point, only the "nqmax" nearest points are used in each of the quadrants around the point.

The special case of (7-21) offers a straightforward method of improving an estimate of the correlation length and thereby improving the covariance function. Suppose the covariance function depends on a defining constant with a known relationship to the correlation length. This is the case in all three examples given: the defining constant in Example 7.1 is the value of k used in (7-11); in Examples 7.2 and 7.3 (where the covariance functions are finite), the defining constant is r_{max}.

Let r_1 be the initial estimate of the correlation length (possibly the MDNN) of a position-dependent variable *f*. Let S_{r_1} be the set of all pairs of data points between $r_1 - \delta r$ and $r_1 + \delta r$ apart. The tolerance δr should be large enough to ensure that S_{r_1} is a meaningful sample. The next step is to calculate

$$C_1 = \text{Avge}\{f_i f_j \text{ such that } (P_i, P_j) \in S_{r_i}\}. \quad (7\text{-}22)$$

The defining constant in the covariance function can be obtained by solving the equation

$$C_1 = C(r_1). \quad (7\text{-}23)$$

The covariance length can then be re-calculated from its relationship with the defining constant.

A more sophisticated method is to calculate the covariances for several distances by sampling, and to use least-squares optimisation to determine the defining constant of the chosen covariance function. This is an adaptation of the method described in Section 9.1 of Cross (1983). The implementation considered here uses the MDNN as the central distance and uses a quarter of its value as a sample-range for each distance. This is illustrated in Figure 7-6.

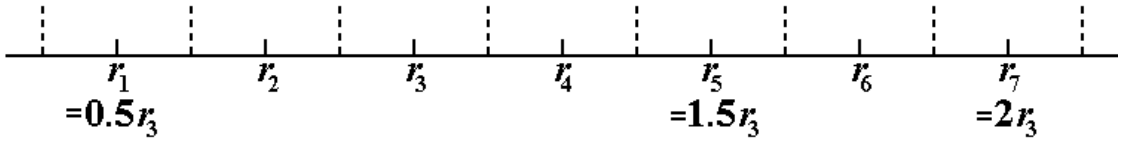


Figure 7-6: Partition of distance-range using MDNN as r_3 and MDNN/4 as each sample-range.

Let S_{r_α} be the set of all pairs of data points between $r_\alpha - \delta r$ and $r_\alpha + \delta r$ apart, where $\delta r = r_3/8$. The next step is to calculate

$$C_\alpha = \text{Avge}\{f_i f_j \text{ such that } (P_i, P_j) \in S_{r_\alpha}\} \text{ for } \alpha = 1, 2, 3, 4, 5, 6, 7. \quad (7\text{-}24)$$

Cross (1983, page 104) suggests fitting a mathematical function to this kind of data, but does not specify how.

The iterative optimisation approach would be to vary the defining constant of the covariance function until that function gives a least-squares fit to the points (r_1, C_1) to (r_7, C_7). This study produced a more direct method, which is as follows.

Let β be the value of α for which $|C_\alpha - 0.5C_0|$ is minimised, meaning that r_β ought to be closer to the correlation length than other values of r_α.

The quantities obtained from (7-24) will give rise to seven equations of the form

$$C(r_\alpha) = C_\alpha. \tag{7-25}$$

Solving (7-25) for each α other than β involves using an inverse covariance formula, such as (7-12) or (7-15) or (7-18), to estimate the defining constant of the transformation model. There will be 6 estimates of that defining constant. Substituting those values in turn into the transformation model, 6 estimates of $C(r_\beta)$ will be obtained in addition to the one already computed. A weighted average of all 7 estimates – with weights based on the size of the sets – should give a much-improved value of C_β since it is based on 7 sets of sample distances rather than one. The appropriate inverse covariance formula is used once more to find the defining constant which ensures that $C(r_\beta) = C_\beta$.

Some caution needs to be exercised in this approach for the following reasons.

- The number of data points may not always be enough for there to be a sufficient number of between-point distances in all of the 7 samples. The method may need to be restricted to a subset of the ranges.
- The quantities computed from (7-24) may vary wildly from the kind of function illustrated in Figures 7-3 to 7-5. If this happens, the assumption that the residual shifts are an errorless signal free of noise should be re-examined.

Another method devised for this study to estimate the extent of correlation is based on both the spread of data points and the behaviour of the signal. This one is only applicable to finite covariance functions because it estimates r_{max} rather than correlation length. For each data point, the nearest data point whose signal is of opposite sign is traced and the furthest data point within that range whose signal has the same sign is noted. The "distance to zero" is estimated by interpolating the two distances. The median "distance to zero" (MDZ) over the full set of data points is computed from the complete set of data points.

Even if statistical methods cannot produce the defining parameter of a covariance model, an intuitive choice – either of the parameter or correlation length – will still allow least-squares collocation to be used as an interpolation tool.

The procedure for estimating the signal at computation points in the special case of (7-21) is described under "Errorless Collocation" in Ruffhead (1987), although it is based on formulae given in Section 7-8 of Heiskanen and Moritz (1967) for a geophysical application. It involves the inverse of a covariance matrix. One advantage of a finite covariance function is that it

leads to a sparse covariance matrix. Using $\overline{\mathbf{s}}$ to denote the predicted signal at computation points,

$$\overline{\mathbf{s}} = \mathbf{Cov}(\overline{\mathbf{s}}, \mathbf{s})\mathbf{Cov}(\mathbf{s}, \mathbf{s})^{-1}\mathbf{s}. \tag{7-26}$$

To provide an element of choice in the way formula (7-26) is applied, it was deemed convenient to convert it into two equations for this study:

$$\overline{\mathbf{s}} = \mathbf{Cov}(\overline{\mathbf{s}}, \mathbf{s})\mathbf{s}_{\mathrm{RV}}, \tag{7-27}$$

where

$$\mathbf{s}_{\mathrm{RV}} = \mathbf{Cov}(\mathbf{s}, \mathbf{s})^{-1}\mathbf{s}. \tag{7-28}$$

The latter term $\mathbf{s}_{\mathrm{RV}}$ is an intermediate vector, and for this study it is labelled "revamped signal vector". Its units are actually the inverse of the units used in the datum shift and the signal (*eg* inverse metres or inverse arc-seconds). Computation of $\mathbf{s}_{\mathrm{RV}}$ does not necessarily require the explicit inversion of a matrix, since it can be treated as the solution of the square system of equations

$$\mathbf{Cov}(\mathbf{s}, \mathbf{s})\mathbf{s}_{\mathbf{RV}} = \mathbf{s}. \tag{7-29}$$

An important property of the revamped signal vector is that, for a given set of control points and a given covariance function, it only needs to be computed once. It is not affected by the choice of computation points.

If least-squares collocation is used purely as a gridding method, providing regularly-spaced datum shifts that are relatively easy to interpolation, then $\mathbf{s}_{\mathrm{RV}}$ only needs to be used once. However, if it used as a model for application at any point or set of points in the area of interest, then $\mathbf{s}_{\mathrm{RV}}$ is worth storing as a part of the model. Equation (7-27) is a far more convenient formula than (7-26).

The case studies in Chapters 8, 12 and 13 have shown that where two or more control points are close together in a cluster, the corresponding elements of s_{RV} may be very large compared to the other elements. This doesn't necessarily lead to large signals in $\overline{\mathbf{s}}$, but it is advisable to predict additional signals in the vicinity of the clusters as a precautionary check.

There will be a different signal for the different components of the datum shift. Any trend model that has been removed to isolate the signal will need to be restored and added to the predicted signal to obtain the predicted datum shift at the computation points.

Noise cannot always be ignored. Although this study has preferred to use (7-20) rather than (7-8) as the basis for least-squares collocation, there may be circumstances where the presence of noise has to be acknowledged. After the removal of the trend model from the measurement, the residual vector at the control points is given by

$$\textit{(centred) measurement} = \textit{signal} + \textit{noise} = \mathbf{s} + \mathbf{n}. \tag{7-30}$$

How much allowance is made for noise depends on how the variance of the residual is split between signal and noise. To minimise the loss of exact interpolation, each $Var(n_i)$ needs to be small compared with each $Var(s_i)$. The covariance matrix of the residuals at the control points, namely $\mathbf{Cov}(\mathbf{s}+\mathbf{n}, \mathbf{s}+\mathbf{n})$, depends on two identities:

$$Var(s_i + n_i) = Var(s_i) + Var(n_i) \tag{7-31}$$

and, since noise is uncorrelated,

$$Cov(s_i + n_i, s_j + n_j) = Cov(s_i, s_j) \text{ if } i \neq j \tag{7-32}$$

In short, the diagonal elements of $\mathbf{Cov}(\mathbf{s}+\mathbf{n}, \mathbf{s}+\mathbf{n})$ use the variance of signal + noise, whereas the off-diagonal elements use the signal variance as C_0 in the chosen covariance formula.

With noise, the matrix equation for the revamped signal becomes

$$\mathbf{Cov}(\mathbf{s}+\mathbf{n}, \mathbf{s}+\mathbf{n})\mathbf{s}_{\mathbf{RV}} = \mathbf{s}+\mathbf{n}, \tag{7-33}$$

while the predicted signal at computation points is (as before) given by

$$\overline{\mathbf{s}} = \mathbf{Cov}(\overline{\mathbf{s}}, \mathbf{s})\mathbf{s}_{\mathrm{RV}}. \tag{7-34}$$

When least-squares collocation is used to obtain datum transformations, the output consists of datum shifts at a set of computation points. To obtain the datum shifts at other points means either repeating the collocation process or interpolating the shifts at the computation points. Some authors deliberately choose regularly-spaced computation points for a one-off application of least-squares collocation. Piecewise bilinear or piecewise bicubic interpolation is then applied to points within the rectangles. Kinneen and Featherstone (2004) use this approach between Australian datums, applying bilinear interpolation for points within the rectangles. This study has not found any examples of smooth bicubic interpolation of residual datum shifts. However, because it makes use of partial derivatives, it has the potential to produce better results and allow wider spacing between grid points.

7.2 Kriging

Kriging is an interpolation technique named after Danie G. Krige. Grgić *et al* (2016) describe it as "a method that constructs a minimum error variance estimation" which "attempts to express trends suggested in the data that can be fitted to any data set by specifying the appropriate variogram model and anisotropy". Bohling (2005b) defines Kriging as "optimal interpolation based on regression against observed z values of surrounding data points, weighted according to spatial covariance values".

Bohling (2005b) introduces three types:

- "Simple kriging", which assumes a constant mean over the entire domain.
- "Ordinary kriging", which assumes the mean is constant in the local neighbourhood of each estimation point.
- "Kriging with a trend", where a linear or higher-order trend is fitted in the neighbourhood of the estimation point. It notes that this used to be known as "universal kriging", but the latter's appearance in other sources indicate that it is still in use.

Grgić *et al* (2016) use Kriging to model distortions in datum shifts for grid coordinates in Croatia after removal of a trend model. They rate it as one of the best methods for that area, along with minimum curvature and inverse distance to a power.

Merry and Whittal (1998) used Kriging to approximate $\Delta\phi$ and $\Delta\lambda$ at grid points $6'$ apart, using the Surfer package. The transformation was from Cape Datum to WGS84 for a test area in South Africa. The version chosen was "universal" Kriging with allowance for a linear trend in $\Delta\phi$ and $\Delta\lambda$. They found that Kriging was faster than least-squares collocation but gave very similar results. Interpolation from the grid was done by cubic splines.

Kriging has similarities to least-squares collocation, including the use of inverse covariance matrices. Dermanis (1984) compared the two methods in general terms, without specifically considering datum shifts, and specified criteria under which the methods coincide. Claims of equivalence to LSC can also be found in Blais (1982), Menz *et al* (2015), and Merry and Whittal (1998). Forsberg and Tscherning (2008) gives "Kriging" as an alternative name for Least Squares Collocation.

One difference from least-squares collocation is the use of semivariance. If S_{r_1} is the set of all pairs of data points between $r_1 - \delta r$ and $r_1 + \delta r$ apart, then the semivariance $\gamma(r)$ is (at least empirically) given by

$$\gamma(r) = 0.5\text{Avge}\left\{\left(f_i - f_j\right)^2 \text{ such that } \left(P_i, P_j\right) \in S_{r_1}\right\}. \tag{7-35}$$

When plotted against lag distance (r), the result is called a semivariogram (or sometimes just a variogram). Examples are shown in Figures 7-7 and 7-8. Both have a "sill", which is the semivariance value at which the semivariogram levels off. The sill may be a maximum or an asymptotic limit. The range of semivariogram models available in Surfer is given in Surfer (2002, pages 181-183).

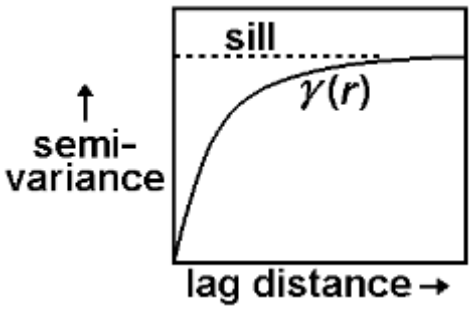


Figure 7-7: Exponential semivariance model.

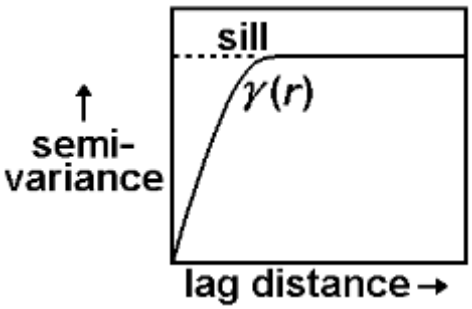


Figure 7-8: Quadratic semivariance model.

Covariance for a given lag distance can be estimated from semivariance using the assumption that the two add up to the variance. They certainly obey that relationship under the condition of second-order stationarity (spatially constant mean and variance). In practice, as noted by Bohling (2005a), the estimated versions of *C(r)* and *γ(r)* will violate that relationship to a greater or lesser extent due to sampling limitations and deviations from second-order stationarity.

7.3 Rubber sheeting

Rubber sheeting is the representation of a surface by conjoining plane triangles; the surface is stretched in a piecewise fashion. A Delaunay triangulation is applied to the control points, which means that the circumcircle of each triangle contains no other control point. In Figure 6-2, the triangles ABD and BCD could be part of a Delaunay triangulation because the angles facing the shared side add up to less than 180°. The triangles ABC and ACD could not, because the angles facing the shared side add up to more than 180°. Interpolation of the data in each triangle is done by the plane triangle that fits the values (considered loosely as "heights") at the vertices.

The question of whether rubber sheeting can be applied to non-triangular polygons, such as quadrilaterals, has not been considered in this study.

Rubber sheeting is a method applicable to datum transformations. It is applied by González-Matesanz *et al* (2003, 2006) to the residual datum shifts after the removal of a 7-parameter conformal model. In so doing, they are conforming to what they describe as "the general strategy followed by Australia, Canada and other countries".

Algorithms for doing a Delaunay triangulation can be found in Lee and Schachter (1980) and Su and Drysdale (1997). Some virtual points are added on the outside (based on local 7PC models) "to avoid excessively sharp triangles near the coast". Working from grid coordinates, an affine transformation of the type described in Section 4.2 is applied to points on the basis of the triangles they belong to.

Rubber sheeting is known by other names. Doytsher (2000) identifies it as an alternative name for "applying a Piecewise Linear Homeomorphic (PLH) transformation". The latter term is also used by Gillman (1985). When applied to transformations by Grgić *et al* (2016), it is referred to as "triangulation with linear interpolation", in keeping with Surfer (2002, pages 135-136).

If the area of interest has a dense coverage of control points (so that the triangles are small) and the residual datum shifts being interpolated are small, the quality of the interpolation will be quite good despite the discontinuities in partial derivatives.

Greaves (2004) describes OSTN02 which takes rubber sheeting one stage further in transforming projected grid coordinates from European Terrestrial Reference System 1989 to Ordnance Survey of Great Britain 1936 (the National Grid). A regular grid of datum shifts is computed by the affine transformations, in this case with a resolution of 1km. This enables straightforward bilinear interpolation which - to some extent - smooths out the gradient discontinuities along the edges of the triangles. Ordnance Survey (2018, 2020) give a similar description of OSTN15, the current national datum transformation between ETRS89 and OSGB89. OSTN15 has been coded into a software application "Grid InQuest II" which is publicly available.

Berk and Komadina (2013) use rubber sheeting to transform local (D48) grid coordinates for Slovenia to ETRS89. A Delauney triangulation was used, based on "virtual tie points" connecting both systems. In this case, virtual tie points were not only created outside the national borders (to enable extrapolation) but replaced directly-observed control points within the country area. "Coordinate shifts on each virtual tie point were determined via best-fit transformation in its immediate neighbourhood."

It would appear that rubber sheeting can be used without converting latitude and longitude to grid coordinates. This is because the triangles can be treated as if they are on a graphical representation of (ϕ, λ) and, as noted in Section 6.1, the affine transformation equations can be expressed in terms of (ϕ, λ).

7.4 Minimum curvature surfaces

Briggs (1974) developed the method of "minimum curvature surfaces" for geophysical data that can be represented by contour maps. The minimum curvature condition leads to differential equations than can be represented approximately by finite difference equations at grid points. For each observation point that does not coincide with a grid point, another difference equation is required for grid points that are vertices of the grid square in which the observation falls. Grid points that coincide with observation points give rise to what are – in effect – boundary conditions. The set of linear algebraic equations is solved iteratively. The approximations computed at the grid points can be interpolated bilinearly.

Dewhurst (1990) applied this approach to a transformation between North American Datums to develop gridded datasets for the shifts $\Delta\phi$ and $\Delta\lambda$, starting from known shifts at irregularly-spaced points. Thus, two mathematical surfaces are prepared for each major region of the country. Computing $\Delta\phi$ and $\Delta\lambda$ for an individual point is achieved with bilinear interpolation based on the 4 surrounding grid points, although Dewhurst does acknowledge other possible interpolation methods. Dewhurst's computer program NADCON – an acronym for North American Datum Conversion - is now available online for a number of datum transformations; see NGS (2013).

NTv2 (National Transformation Version 2) is another computer program using minimum curvature for datum transformations. It is similar to NADCON although originally designed for Canada. A key difference is that NTv2 allows for sub-grids with a higher density than the main grid. See Junkins and Farley (1995).

NADCON's use of minimum curvature surfaces appears to be a composite method only in the sense that gridded datasets are created and then interpolated. The application of minimum curvature as described in Grgić *et al* (2016), however, is to residual datum shifts after the removal of a trend model. It is noted (Ibid, Table 1) that the process "produces a smooth surface and, consequently, source data is not always treated exactly", confirming that interpolation of the gridded datasets will not necessarily produce the original datum shifts at the observed points exactly.

There is no obvious reason why minimum curvature surfaces cannot be applied to geodetic coordinates rather than projected grid coordinates.

7.5 Inverse distance to a power

Inverse distance weighting is a scattered-data interpolation algorithm proposed by Shepard (1968). It is very easy to implement. The weights are non-negative quantities whose sum is 1, and this ensures the interpolating function never strays outside the range of the interpolated values.

A continuous and smooth function *f* is interpolated at scattered control points $\{P_i\}$ by the formula

$$f_P = \frac{\sum f_i / \|P - P_i\|^n}{\sum 1 / \|P - P_i\|^n} \tag{7-36}$$

where $\|P - P_i\|$ is the distance from *P* to P_i. The formula takes the limiting value f_j when *P* coincides with a control point P_j.

An alternative form of (7-36), which avoids zero divisors and processing of large numbers, is

$$f_P = \frac{f_j + \sum_{i \neq j} f_i \cdot \left(\|P - P_j\| / \|P - P_i\|\right)^n}{1 + \sum_{i \neq j} \left(\|P - P_j\| / \|P - P_i\|\right)^n} \tag{7-37}$$

where

$$j = \text{value of } i \text{ that minimises } \|P - P_i\|. \tag{7-38}$$

The value of *n* in formula (7-36), and hence in (7-37), must be at least 2 to ensure smooth interpolation. If *n* is exactly 2, the control points are said to be weighted by inverse square distances. One argument for having *n* greater than 2 is that it limits the influence of distant points.

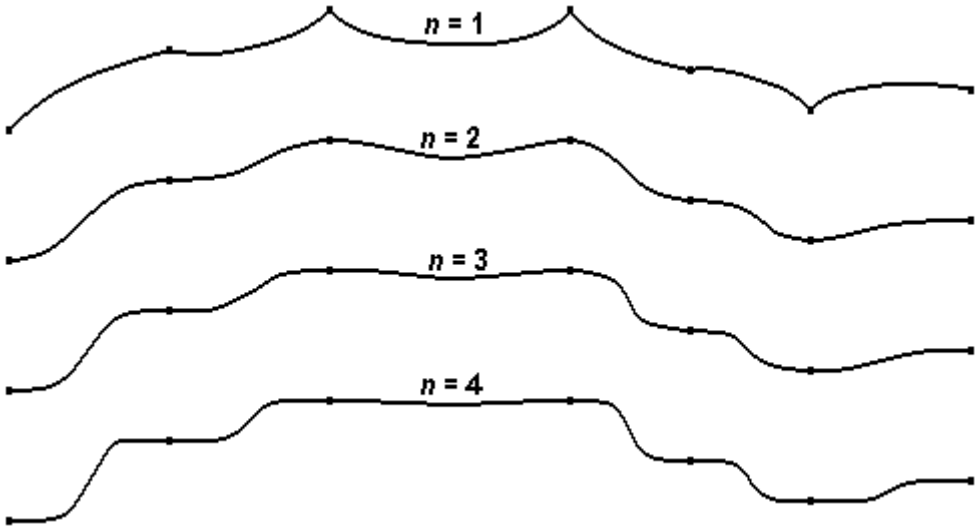


Figure 7-9: Characteristics of inverse distance interpolation using powers 1, 2, 3 and 4.

Figure 7-9 illustrates the effect of inverse distance interpolation on data which depends on a single variable. None of the data fits is totally satisfactory. Increasing the power *n* reduces the dip between the 3rd and 4th points, but accentuates the changes in curvature between other control points.

Setting aside the case where the power *n* is 1, the main drawback of inverse distance weighting is that it imposes zero gradients at the control points (unless the power is 1). When it is used to fit a surface, it has the tendency to generate concentric contours around the control points. This is described by several authors, among them Attaouia *et al* (2017) and Musashi *et al* (2018), as a bullseye effect.

A variation of (7-36) can be applied to control points which have weights based on their perceived reliability:

$$f_P = \frac{\sum w_i f_i / \|P-P_i\|^n}{\sum w_i / \|P-P_i\|^n} \tag{7-39}$$

Control points with the highest weights will influence the interpolated *f* within a larger local radius than control points with the lowest weights.

Grgić *et al* (2016) apply inverse-distance-to-a-power to interpolate residual transformations. Given that the residual datum shifts are smaller than the original datum shifts, the drawback of zero gradients at control points is less of a problem than in it would be if the method had been applied to the original datum shifts. When inverse square distances were used ($n = 2$),

the accuracy over Croatia was comparable to that achieved by Kriging and minimum curvature.

Grgić *et al* (2016) also mentions that one possible use of "inverse distance to a power" is as a smoothing interpolator rather than exact interpolator. However, they chose to use the method as an exact interpolator, setting the smoothing parameter to zero.

One possible smoothing interpolator is the formula

$$\bar{f}_P = \frac{\sum f_i / \left(\sqrt{\delta^2 + \|P - P_i\|^2}\right)^n}{\sum 1 / \left(\sqrt{\delta^2 + \|P - P_i\|^2}\right)^n} \tag{7-40}$$

where δ is a positive constant and $\|P - P_i\|$ is the distance from P to P_i. The control points closest to point P will have the greatest influence on the value of $\overline{f}_P$ but $\overline{f}$ will not interpolate the control points exactly. One characteristic of (7-40) is that the numerator and denominator will always be finite. δ is a "smoothing parameter" and setting it to zero is equivalent to turning (7-40) into (7-36), which is an exact interpolator. Formula (7-40) matches that on page 115 of Surfer (2002), so it must be the means by which Surfer applies the smoothing parameter δ.

An alternative smoothing interpolator is the formula

$$\bar{f}_P = \frac{\sum f_i / [\delta + \|P - P_i\|]^n}{\sum 1 / [\delta + \|P - P_i\|]^n}, \tag{7-41}$$

where δ is a positive constant which acts as a smoothing parameter. The properties of formula (7-41) are much the same as those of (7-40). Formula (7-41) matches that given by Tomczak (1998) which claims Keckler (1995) as its source. The latter is a user guide to Surfer Version 6 (1995), so it may have been superseded by Surfer (2002).

It is worth mentioning that in the case of inverse square distance (n=2), formula (7-40) simplifies to

$$\bar{f}_P = \frac{\sum f_i / \left[\delta^2 + \|P - P_i\|^2\right]}{\sum 1 / [\delta^2 + \|P - P_i\|^2]}. \tag{7-42}$$

According to Surfer (2002, page 115), the smoothing factor parameter used in (7-40) allows the user to incorporate an "uncertainty" factor associated with the input data: "the larger the smoothing factor parameter, the less overwhelming influence any particular observation has in computing a neighbouring grid".

There remains the issue of how the smoothing parameter δ used in formula (7-40) is selected. Golden Software Support implies that selection is a trial-and-error process to remove the bullseye effect without making the surface too smooth: "For example, instead of 0, you could try a smoothing value of 10. If that is too smooth, you can step the smoothing to 5 to see if that is better." (Woodson, 2016).

The method of "inverse distance to a power" can be used to generate a rectangular mesh of points which can be interpolated with bilinear or bicubic functions. This may be unnecessary because of the simplicity of the inverse-distance method itself.

As noted earlier, the relative ease of computing distances from geodetic coordinates means that it is not essential to apply a map projection to the coordinates of data points.

7.6 Radial basis functions

A radial basis function is a real-valued function whose value depends only on some measure $\|\ldots\|$ of the distance from the origin. It can be applied to the same measure of the distance between two points P and Q, which will be denoted by $\|P - Q\|$. Following Baxter (1992) and Attaouia *et al* (2017), the radial basis function is denoted here by the symbol φ, rather than the symbol ϕ used by Powell (2005) and Bullinaria (2015a, 2015b).

Interpolation based on radial basis functions is based on the formula

$$f_P = \sum_i w_i \varphi(\|P - P_i\|) \tag{7-43}$$

where the points P_i are control points where the function being interpolated is known.

For many choices of φ, the parameters w_i can be obtained by solving the square system of equations

$$f_i = \sum_j w_j \varphi(\|P_i - P_j\|). \tag{7-44}$$

Surfer offers five possible radial basis functions for interpolating residual datum shifts. They are as follows.

- Inverse multiquadric:

$$\varphi(r) = 1/\sqrt{1 + (\varepsilon r)^2} \text{ or } \varphi(r) = 1/\sqrt{r^2 + R^2}. \tag{7-45}$$

- Multilog:

$$\varphi(r) = \ln(r^2 + R^2). \tag{7-46}$$

- Multiquadric, sometimes called Hardy multiquadric arising from Hardy (1990):

$$\varphi(r) = \sqrt{1 + (\varepsilon r)^2} \text{ or } \varphi(r) = \sqrt{r^2 + R^2}. \quad (7\text{-}47)$$

- Natural cubic spline:

$$\varphi(r) = (1 + (\varepsilon r)^2)^{3/2} \text{ or } \varphi(r) = (r^2 + R^2)^{3/2}. \quad (7\text{-}48)$$

- Thin plate spline:

$$\varphi(r) = (r^2 + R^2)\ln(r^2 + R^2). \quad (7\text{-}49)$$

Surfer (2002, page 132) comments "In terms of the ability to fit your data and to produce a smooth surface, the Multiquadric method is considered by many to be the best. All of the radial basis function methods are exact interpolators, so they attempt to honour your data. You can introduce a smoothing factor to all the methods in an attempt to produce a smoother surface." In all 5 cases listed above, R^2 is described as a shaping or smoothing parameter (ibid, page 133).

Confusingly, Surfer (2002) says that the methods which are exact interpolators include "Radial Basis Function when you do not specify an R^2 value" (ibid, page 156). The author of that overview is in conflict with the author of the radial basis description (ibid, pages 132-135), with Bullinaria (2015a) and with Hardy (1990). The choice of R^2 influences the shape of the interpolating surface, but the surface obtained from (7-43) and (7-44) is still an exact fit at the control points. For this reason, the term "shaping parameter" is preferable to the misleading "smoothing parameter".

Surfer (2002, page 133) comments "There is no universally accepted method for computing an optimal value for this factor. A reasonable trial value for R^2 *Parameter* is between the average sample spacing and one-half the average sample spacing." Surfer's default value for R^2 (ibid, page 134) is given in (7-50) below; it has been adopted, for example, by Attaouia *et al* (2017).

$$R^2_{\text{default}} = (\text{length of diagonal of the data extent})^2/(25 \times \text{number of data points}). \quad (7\text{-}50)$$

From (7-43), it is clear that the basis functions $\varphi(\|P - P_j\|)$ and the weights w_j are as numerous as the control points. If there are a large number of control points then the solution of (7-44) becomes computationally costly (although it is a one-off exercise). Various approaches have been suggested to address this problem:

- Lazzaro and Montefusco (2002) proposed a variation that decomposes the interpolation problem into several steps. This sacrifices the conceptual simplicity of the basic method for a more elaborate process.

- Bullinaria (2015a) suggests forming the basis functions from a much smaller subset of the control points. Bullinaria (2015b) describes ways in which this can be done.
- Surfer (2002, pages 104-106) describes ways of limiting the number of control points to include in the interpolation at each grid node. Surfer software provides options for specifying a search ellipse (defining the size of the local neighbourhood) and search rules (that determine the number of points to actually consider within the neighbourhood). This approach is actually obligatory if there are more than 750 control points.

For this study, the implementation of radial basis functions involves the expression of a variable datum shift as the sum of a trend model and a signal, by analogy with (7-20) which is used in least-squares collocation. The radial basis functions can be applied to interpolate the residuals from the trend model.

With this approach, solving (7-44) for $w_1, w_2, w_3, \ldots$ is analogous to solving (7-29) for vector $\mathbf{s}_{\mathrm{RV}}$. The vector consisting if the w_i can even be regarded as a revamped signal vector and denoted by $\mathbf{s}_{\mathrm{RV}}$.

An important property of the revamped signal vector is that, for a given set of control points and a given radial basis function, it only needs to be computed once. It is not affected by the choice of computation points.

The analogy with least-squares collocation can be extended to an RBF-equivalent of the covariance matrix. The radial basis function arising from points *i* and *j* is $\varphi(r_{i,j})$ where $r_{i,j}$ is the distance between points *i* and *j*. A radial basis matrix, denoted in this thesis by $\mathbf{Rdl}(\mathbf{s}_{\mathrm{A}}, \mathbf{s}_{\mathrm{B}})$, extends this to two sets of signals (not necessarily distinct but not necessarily equal in size).

The RBF-equivalents of (7-27) and (7-28) are (7-51) and (7-52), given below. For computation points, the predicted signals are given by

$$\overline{\mathbf{s}} = \mathbf{Rdl}(\overline{\mathbf{s}}, \mathbf{s})\mathbf{s}_{\mathrm{RV}}, \tag{7-51}$$

where

$$\mathbf{s}_{\mathrm{RV}} = \mathbf{Rdl}(\mathbf{s}, \mathbf{s})^{-1}\mathbf{s}. \tag{7-52}$$

Whether the inverse matrix needs to be computed explicitly depends on whether the equation

$$\mathbf{s} = \mathbf{Rdl}(\mathbf{s}, \mathbf{s})\mathbf{s}_{\mathrm{RV}} \tag{7-53}$$

can be solved by some other means. If $\mathbf{Rdl}(\mathbf{s}, \mathbf{s})$ is positive definite, then solution by Cholesky decomposition is feasible. The implementation of Cholesky for this study is described in sub-appendix G.2.2.

If a radial basis function is used purely as a gridding method, providing regularly-spaced datum shifts that are relatively easy to interpolate, then $\mathbf{s}_{\mathrm{RV}}$ only needs to be used once. However, if it used as a model for application at any point or set of points in the area of interest, then $\mathbf{s}_{\mathrm{RV}}$ is worth storing as a part of the model. Putting it another way, storing the quantities w_i makes (7-44) easy to use and re-use to predict signal components of datum shifts.

7.7 Nearest neighbour interpolation

Grgić *et al* (2016) includes this possible method for interpolating residual shifts at the control points to generate a grid. The interpolated residual shifts at each grid point come from the nearest control point. Arya *et al* (1998) is quoted as a reference; it is not essential for implementing the interpolation method, but it does offer time-saving methods of finding the nearest control point when the original data set is large.

The methodology of Arya *et al* (1998) for computing minimum distances is applicable to $\mathbb{R}^3$ as well as to $\mathbb{R}^2$; therefore, even if that is used, it is not essential to apply a map projection to the coordinates of control points.

As nearest neighbour selects a data-point function value for each grid point, (with no attempt to compute in-between function values), this interpolation method is not very sophisticated, especially when compared with methods that generate a smooth surface through the control points. Because of this, bilinear interpolation rather than bicubic interpolation is the obvious method to apply within the rectangles of the grid.

Figure 7-10 illustrates the discontinuities of nearest neighbour interpolation. The left-hand diagram shows just two of the patches over which the interpolating function is constant.

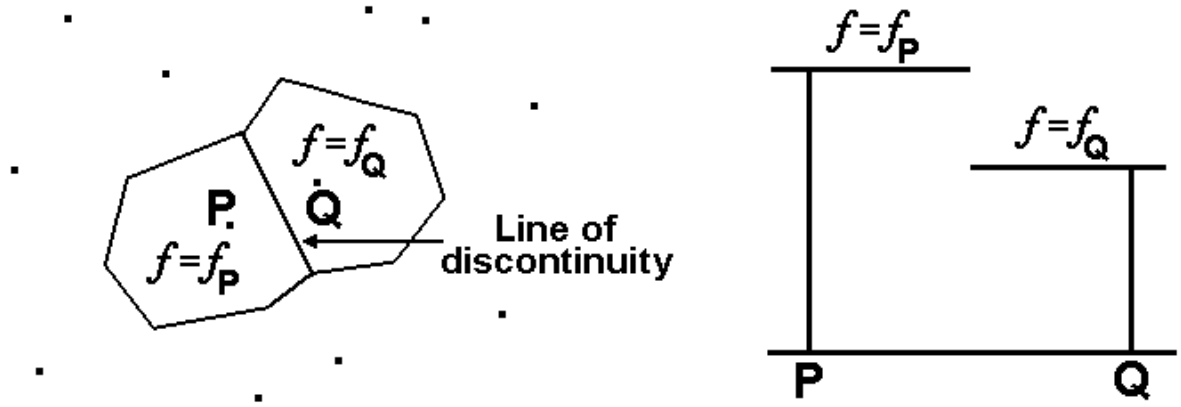


Figure 7-10: Nearest neighbour interpolation over the areas around control points P & Q and along the line PQ.

One characteristic of nearest neighbour interpolation is that all data-point values will recur at grid points except where a control point of enclosed by 4 grid points that are each closer to another control point. This means that the peaks and troughs of the function values at the control points almost always appear in the computation points. In this respect it has an advantage over inverse distance to a power, for example, where dips occur between peaks (as illustrated in Figure 7-9) and ridges occur between troughs.

7.8 Natural-neighbour interpolation

This interpolation method was proposed by Sibson (1981). Alternative names for the method are Sibson Interpolation and "Area-Stealing" (Musashi *et al*, 2018). For each point of interest P, it computes a weighted average of function values at a selection of neighbouring control points.

The way it does this is by the use of "Voronoi cells", illustrated in Figure 7-11. A one-off tessellation of the region of interest creates a Voronoi cell $V(P_i)$ for each control point P_i which consists of all points nearer to that control point than to any other control point. (The sides of each cell are perpendicular bisectors of chords joining control points.) For the point P where an interpolated value is sought, the Voronoi cell $V(P)$ consists of all points nearer to P than to any control point. For each cell $V(P_i)$ that intersects with $V(P)$, the overlapping area for $V(P) \cap V(P_i)$ is computed. The weights w_i assigned to the function values f_i are proportional to those areas, and the sum of the weights is 1.

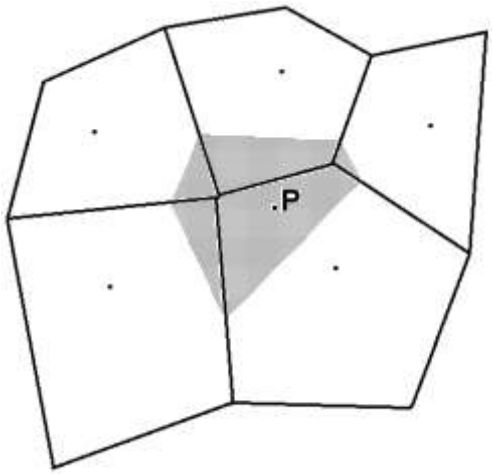


Figure 7-11: Example of Voronoi cells around the point P; the areas of the shaded overlaps (clockwise from top) are in the ratio 58:4:109:30:16 so the weights applied to the 5 nearby control points are 0.27, 0.02. 0.50, 0.14 and 0.07.

This provides a smoother approximation to the underlying function than nearest-neighbour interpolation. There is a substantial amount of computation involved, but it does avoid the need to solve a large system of equations.

Grgić *et al* (2016) includes natural-neighbour interpolation among the methods applied for interpolating residual shifts at the control points to generate a grid. It seems entirely possible that this method can be applied without a map projection, at least in non-polar regions, using approximations of distances and areas. It is accepted, however, that this should be investigated rather than assumed.

7.9 Modified Shepard's method

Although Shepard (1968) considered variations on inverse distance to a power, "Modified Shepard's method" usually describes a variation proposed by other authors since then.

The version used in Surfer was proposed by Franke and Nielson (1980). It is sometimes called the modified quadratic Shepard's method. It is based on a local quadratic polynomial fit in the neighbourhood of each control point. The procedure continues as an inverse distance model using surface values obtained from the fitted quadratic surface rather than the original control points. The method can be exact or approximate, depending on whether a smoothing factor is specified.

The Surfer implementation uses a full sector search as described in Renka (1988). The local neighbourhood for the quadratic fit is just large enough to include a user-specified number of local neighbours (the "quadratic neighbours" parameter). The interpolated values are generated using a distance-weighted average of the previously-computed quadratic fits

associated with neighbouring observations. The user-specified "weighting neighbours" parameter defines a circular neighbourhood just large enough to contain that number of neighbours.

Grgić *et al* (2016) applied the modified quadratic Shepard's method to interpolate residual datum shifts at grid points. Whether it was used as an exact interpolator is not stated, but it seems likely that was the case. There is no mention in the paper of a smoothing parameter (although Surfer does provide the facility), and they had opted for a smoothing parameter of zero when using inverse distance to a power.

7.10 Moving least squares

Moving least squares is a method of generating surfaces proposed by Lancaster and Salkauskas (1981). It can be used for smoothing or interpolating scattered data. For any given grid point, the function is approximated using the values of control points within a search ellipse to compute a weighted average. If there are fewer than the specified minimum number of control points within the search ellipse, the grid point is blanked.

Grgić *et al* (2016) examined moving least squares, which Surfer prefers to call "moving average", as a method of interpolating residual shifts at the control points to generate a grid. Their results appear to show that this was the least effective method. That conclusion is questionable, however, because the diagram of remaining positional distortions for moving average resembles the reverse of the no-distortion-model (suggesting that twice the required correction has been applied).

7.11 Polynomial regression

This method is used to define large-scale trends and patterns in source data and is not an interpolator. Surfer allows powers of x up to 4 and powers of y up to 4 with a maximum total order of 4. Grgić *et al* (2016) used this for approximating residual shifts to generate a grid, specifying 3 as a maximum power for x, 3 as a maximum power for y, and 4 as a maximum total order.

This method has similarities to multiple regression equations. However, Surfer includes all the permitted combinations of the x and y components in the polynomial equation, so there is no attempt to eliminate statistically-insignificant terms.

7.12 Local polynomial

This method was proposed by Fan and Gijbels (1996). It assigns values to the grid nodes by using the weighted least-squares fit with data within each grid node's search ellipse.

The results obtained from this method by Grgić *et al* (2016) were only slightly inferior to those obtained by Kriging, minimum curvature and inverse distance to a power.

7.13 Hybrid inverse power function embodying accelerated decline (HIPFEAD)

This is an original method devised for this study. It involves a new process for calculating weights of function values for distance-related interpolation. It resembles inverse distance to a power (see Section 7.5). However, it imposes a limit-of-influence, removing the influence of control points beyond a given distance from the point of interest. It does this by a smooth join between inverse distance to a power and a low-degree polynomial function of distance. The latter accelerates the decline of the weighting function, hence its name.

Two subtypes of HIPFEAD were considered in this project:

- Hybrid inverse square function embodying accelerated decline (HISFEAD), in which inverse square distance is joined smoothly to a low-degree polynomial.
- Hybrid inverse cubic function embodying accelerated decline (HICFEAD), in which inverse cubic distance is joined smoothly to a low-degree polynomial.

7.13.1 Hybrid inverse square function embodying accelerated decline (HISFEAD)

The general version of the weighting function has two parameters:

- The distance r_j which indicates where the inverse square function joins the polynomial.
- The limit-of-influence r_{max} which indicates the distance beyond which the function weighting is zero.

The smoothness and continuity requirements at $r = r_j$ and $r = r_{max}$ constitute 4 conditions. As a result, the low-order polynomial $g(r)$ will (in general) be cubic. The weighting function is illustrated in Figure 7-12. The continuation of $1/r^2$ is included to show the extent $g(r)$ deviates from it.

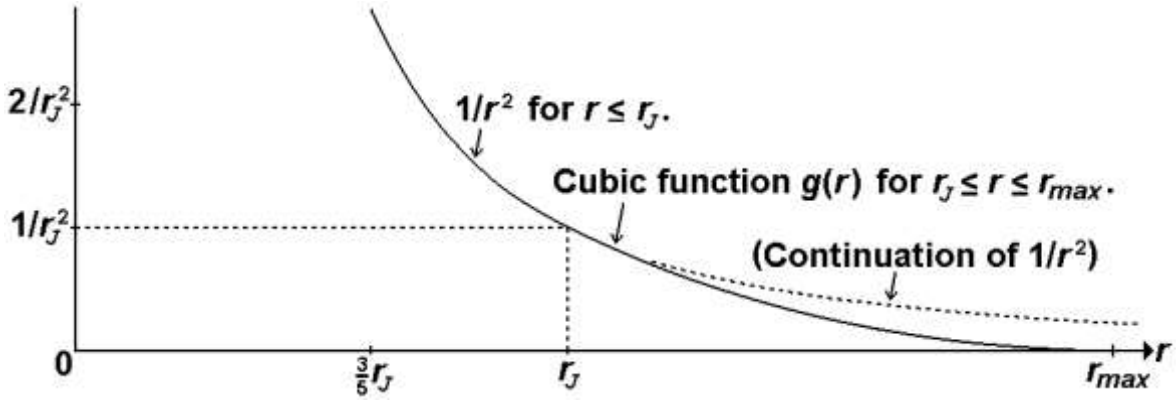


Figure 7-12: General form of the hybrid inverse square function with parameters r_J and r_{max}.

For the weighting of function values at control points on a surface around point *P*, the effect can best be illustrated by Figure 7-13. The circles defined by $r = r_J$ and $r = r_{max}$ can be regarded as "radial partitioning" of the area of interest. The unlabelled dots are illustrative control points.

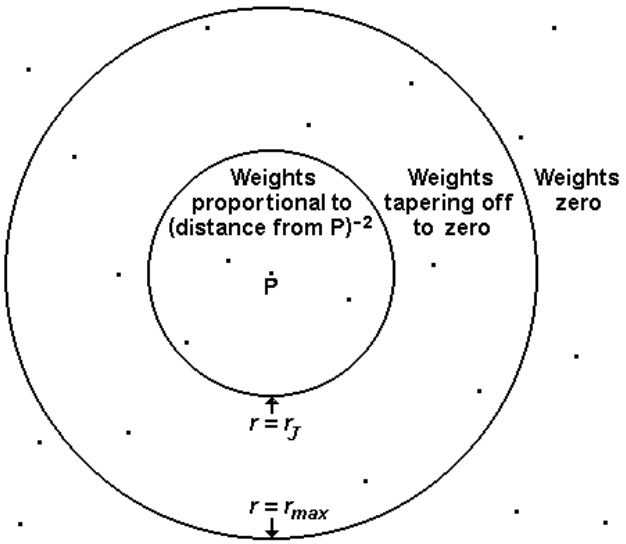


Figure 7-13: Effect of radial partitioning on the weights defined by the hybrid inverse square function for interpolation at *P*.

To achieve the smooth join with the *r*-axis at $r = r_{max}$, the cubic function must take the form

$$g(r) = (a - br)(r_{max} - r)^2. \tag{7-54}$$

It follows that

$$g(r) = a(r_{max}^2 - 2r_{max}r + r^2) - b(r_{max}^2 r - 2r_{max}r^2 + r^3) \tag{7-55}$$

and

$$g'(r) = a(-2r_{max} + 2r) - b(r_{max}^2 - 4r_{max}r + 3r^2). \tag{7-56}$$

Since $g(r_1) = 1/r_J^2$ and $g'(r_J) = -2/r_J^3$, it follows that

$$a(r_{max}^2 - 2r_{max}r_J + r_J^2) - b(r_{max}^2 r_J - 2r_{max}r_J^2 + r_J^3) = 1/r_J^2 \tag{7-57}$$

and

$$a\left(-2r_{max}+2r_J\right)-b\left(r_{max}^2-4r_{max}r_J+3r_J^2\right)=-2/r_J^3. \tag{7-58}$$

Multiplying (7-57) by 2 and (7-58) by $-\left(r_{max}-r_J\right)$ produces the following:

$$a\left(2r_{max}^2-4r_{max}r_J+2r_J^2\right)-b\left(2r_{max}^2r_J-4r_{max}r_J^2+2r_J^3\right)=2/r_J^2 \tag{7-59}$$

and

$$a\left(2r_{max}^2-4r_{max}r_J+2r_J^2\right)+b\left(r_{max}^3-5r_{max}^2r_J+7r_{max}r_J^2-3r_J^3\right)=\left(2r_{max}-2r_J\right)/r_J^3. \tag{7-60}$$

.

Subtracting (7-59) from (7-60),

$$b\left(r_{max}^3-3r_{max}^2r_J+3r_{max}r_J^2-r_J^3\right)=\left(2r_{max}-4r_J\right)/r_J^3, \tag{7-61}$$

and hence

$$b=\frac{2r_{max}-4r_J}{r_J^3\left(r_{max}-r_J\right)^3}. \tag{7-62}$$

From (7-57),

$$a\left(r_{max}^2-2r_{max}r_J+r_J^2\right)=b\left(r_{max}^2r_J-2r_{max}r_J^2+r_J^3\right)+1/r_J^2. \tag{7-63}$$

Applying (7-62),

$$a\left(r_{max}^2-2r_{max}r_J+r_J^2\right)=\frac{2r_{max}-4r_J}{r_J^2\left(r_{max}-r_J\right)}+1/r_J^2, \tag{7-64}$$

which can be rewritten as

$$a\left(r_{max}-r_J\right)^2=\frac{2r_{max}-4r_J+r_{max}-r_J}{r_J^2\left(r_{max}-r_J\right)}=\frac{2r_{max}-4r_J+r_{max}-r_J}{r_J^2\left(r_{max}-r_J\right)}. \tag{7-65}$$

Therefore

$$a=\frac{3r_{max}-5r_J}{r_J^2\left(r_{max}-r_J\right)^3}. \tag{7-66}$$

If $r_{max}<2r_J$, then the transition from $1/r^2$ to zero is relatively abrupt. If $r_{max}>2r_J$, then the change from $1/r^2$ to zero is more gradual but the objective of limiting the number of points used in the interpolant is compromised. Setting $r_{max}=2r_J$ would seem a good compromise. It has the additional attraction that the values of a and b are $1/r_J^4$ and 0 respectively, making $g(r)$ a quadratic function:

$$g(r)=\left(2r_J-r\right)^2/r_J^4=16(r_{max}-r)^2/r_{max}^4. \tag{7-67}$$

This choice of polynomial is adopted as the one which defines hybrid inverse distance function embodying accelerated decline (HISFEAD) unless preceded by the word “generalised”.

Figures 7-14 and 7-15 show the two ways of illustrating HISFEAD graphically, each based on a single defining constant.

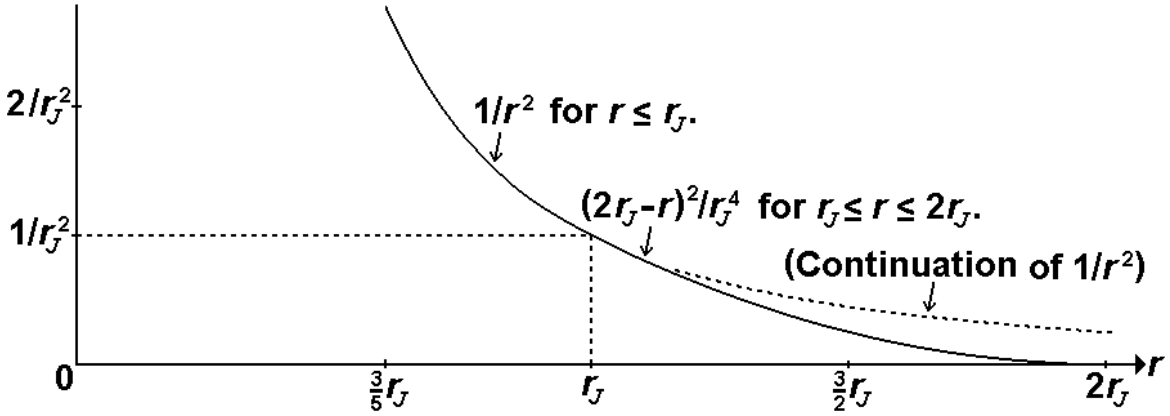


Figure 7-14: HISFEAD with r_J as the defining constant.

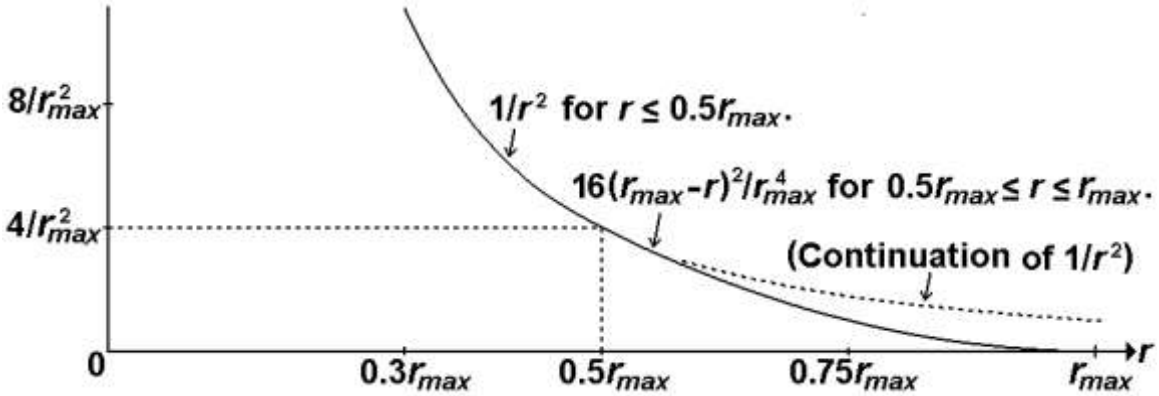


Figure 7-15: HISFEAD with r_{max} as the defining constant.

The basic HISFEAD interpolation formula is

$$f_P = \frac{\sum w_i f_i}{\sum w_i} \tag{7-68}$$

where

$$w_i = \begin{cases} 1/d_i^2 & \text{if } 0 \le d_i \le 0.5r_{max}; \\ 16(r_{max} - d_i)^2/r_{max}^4 & \text{if } 0.5r_{max} \le d_i \le r_{max}; \\ 0 & \text{if } d_i \ge r_{max} \end{cases} \tag{7-69}$$

and

d_i = distance from P to control point i. (7-70)

The values of w_i are all non-negative, as are the values of the normalised weights $w_i/(\sum w_i)$.

The application of HISFEAD has a similar problem to inverse distance to a power, namely a near-zero divisor when the point of interest is close to a control point. In (7-69), $w_i = \infty$ when $d_i = 0$.

The solution is similar to that used in (7-37) and (7-38).

Let j = value of i that minimises d_i. (7-71)

If $d_j \geq 0.5r_{max}$ the proximity problem does not arise. If $d_j < 0.5r_{max}$, which means the reciprocal of w_j is d_j^2, then (7-68) can be replaced by

$$f_P = \frac{f_j + \sum_{i \neq j} d_j^2 w_i f_i}{1 + \sum_{i \neq j} d_j^2 w_i}. \quad (7\text{-}72)$$

This avoids the need to compute w_j.

One potential characteristic of HISFEAD is the interpolant taking a constant value over one or more sub-areas. This will happen if there is an area for which for which only one particular control point is within distance r_{max}; the interpolant will take the value of f at that control point for the whole of that area. (This is, of course, a characteristic of all sub-areas in the case of nearest-neighbour interpolation.)

A problem arises if a point of interest is more than r_{max} from *every* control point. There are two would-be solutions, but each of them is problematic:

- Setting the interpolated f to zero in sub-areas which are at least r_{max} from all control points; but this introduces discontinuities at the boundaries of those sub-areas.
- Setting the interpolated f to nearest neighbour in sub-areas which are at least r_{max} from all control points; but this introduces discontinuities across lines which are equidistant from two control points.

It is therefore a necessary prerequisite that r_{max} is sufficiently large to ensure that *every* possible point of interest is within r_{max} of at least one control point.

HISFEAD can be used locally to avoid one of the problems with surface-fitting methods that are not strictly bounded (NSB), such as least-squares collocation and radial basis functions. An interpolating fit can be volatile due to two or more control points being too close together, as is the case with points R and S in Figure 7-2. There may be more than one such cluster of points in the dataset.

The solution is to create a pseudo control point for each cluster, using the average of the coordinates and the average f. The NSB method as modified by HISFEAD consists of the following stages.

- The NSB method is applied to an interim dataset consisting of the pseudo control points and the control points outside the clusters.
- The residuals at the cluster points are computed (from the known values of *f* minus the values of the NSB approximation).
- For each cluster, a local limit-of-influence r_{max} is set to the minimum distance to other control points (or a smaller value) and a HISFEAD function is defined to interpolate the residuals at the cluster's control points.
- The modified NSB function is the NSB function plus the sum of the HISFEAD functions.

The above function interpolates all the control points and has C^1 continuity. However, the strictly bounded property of HISFEAD removes the danger of volatility from the clusters.

7.13.2 Hybrid inverse cubic function embodying accelerated decline (HICFEAD)

HICFEAD is analogous to HISFEAD, but it is a modification of inverse distance weighting to the power 3. The formulae are (7-68) to (7-70) except that (7-69) is replaced by (7-73) below.

$$w_i = \begin{Bmatrix} 1/d_i^3 & \text{if } 0 \le d_i \le 0.5r_{max}; \\ 64(r_{max} - d_i)^3/r_{max}^6 & \text{if } 0.5r_{max} \le d_i \le r_{max}; \\ 0 & \text{if } d_i \ge r_{max} \end{Bmatrix} \quad (7\text{-}73)$$

The function on which the weights are based is illustrated in Figure 7-16. It is easily shown that the joins at $0.5r_{max}$ and r_{max} are smooth. The smoothness of the function ensures that HICFEAD generates a C^1 surface.

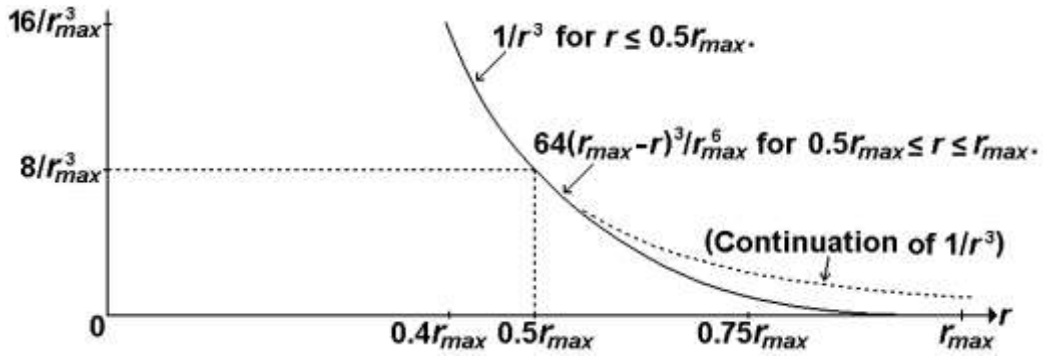


Figure 7-16: HICFEAD with r_{max} as the defining constant.

The application of HICFEAD has a similar problem to inverse distance to a power, namely a near-zero divisor when the point of interest is close to a control point. In (7-73), $w_i = \infty$ when $d_i = 0$.

The solution is similar to that used in (7-37) and (7-38).

Let j = value of i that minimises d_i. (7-74)

If $d_j \geq 0.5r_{max}$ the proximity problem does not arise. If $d_j < 0.5r_{max}$, which means the reciprocal of w_j is d_j^3, then (7-68) can be replaced by

$$f_P = [f_j + \sum_{i \neq j} d_j^3 w_i f_i] \,/\, [\,1 + \sum_{i \neq j} d_j^3 w_i].$$ (7-75)

This avoids the need to compute w_j.

As with HISFEAD, and for the same reasons, HICFEAD requires that r_{max} to be sufficiently large to ensure that *every* possible point of interest is within r_{max} of at least one control point.

7.14 Linear interpolation variant on nearest neighbour (LIVONN)

It was noted in Section 7.7 that the nearest neighbour method selects a data-point function value for each grid point, with no attempt to compute in-between function values.

This thesis introduces the new method "linear interpolation variant on nearest neighbour" (LIVONN). This includes a mechanism for computing intermediate function values when the distances to the nearest control point and 2nd-nearest control point are of similar magnitude. The effect is illustrated in Figure 7-17. The flat areas (the patches over which the interpolating function is constant) are considerably reduced in size from those in Figure 7-10. There is still a probability that the peaks and troughs at the control points will appear among the computation points, but there is far greater continuity than with nearest neighbour.

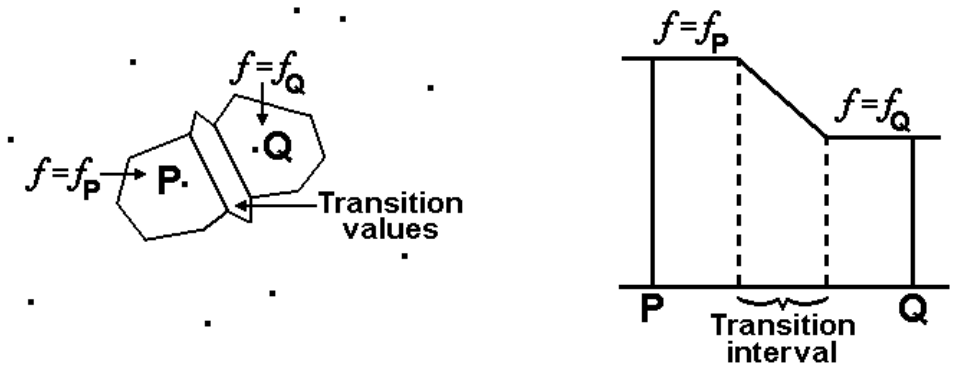


Figure 7-17: The Effect of LIVONN over the areas around control points P & Q and along the line PQ.

The discontinuity of nearest-neighbour interpolation is not completely eliminated. Figure 7-18 gives an example of a discontinuity between the lines bisecting ∠PMQ and ∠RMS. It only

occurs because in this instance there are no control points inside the circle through P, Q, S and R. (M is the centre of that circle and is not a control point.)

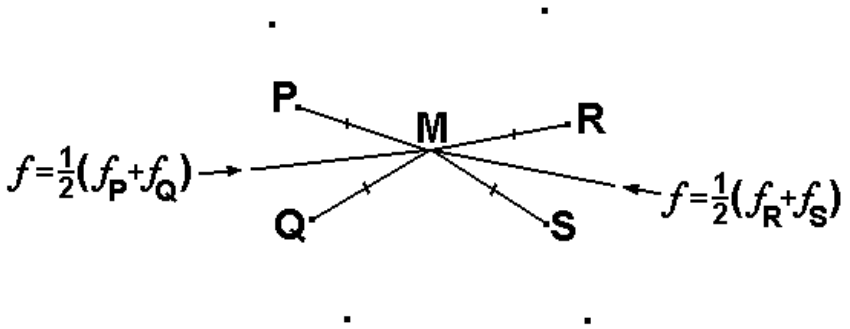


Figure 7-18: Example of LIVONN discontinuity at a point M which is equidistant from control points P, Q, R & S and is further from all other control points.

The earlier description of LIVONN specified that the distances to the nearest and 2nd-nearest control points having "similar magnitude". To make this more precise, it is necessary to define a "transition interval" of the sort used in Figure 7-17. A transition interval is the fraction of an interval over which the interpolation is linear. Either side of that interval, nearest neighbour applies.

The transition interval is a parameter of LIVONN. If it is expressed by the identity

$$\text{transition interval} = 1 - 2\alpha, \tag{7-76}$$

then α:(1-2α):α represents the partitioning of the interpolation interval into constant (nearest neighbour), linear transition and constant (nearest neighbour).

If P_1 and P_2 are (respectively) the nearest and 2nd-nearest control points to P, then

$$f_P = \left\{ \begin{matrix} f_{P_1} & \text{if } \|P-P_1\| \le \alpha\|P-P_2\|; \\ f_{P_1} + (f_{P_2} - f_{P_1})(\|P-P_1\| - \alpha\|P-P_2\|)/[(2-2\alpha)\|P-P_2\|] & \\ & \text{if } \|P-P_1\| > \alpha\|P-P_2\|. \end{matrix} \right\} \tag{7-77}$$

It is easily verified that the latter expression satisfies the requirement that

$$f_P = 0.5(f_{P_1} + f_{P_2}) \text{ when } \|P-P_1\| = \|P-P_2\|. \tag{7-78}$$

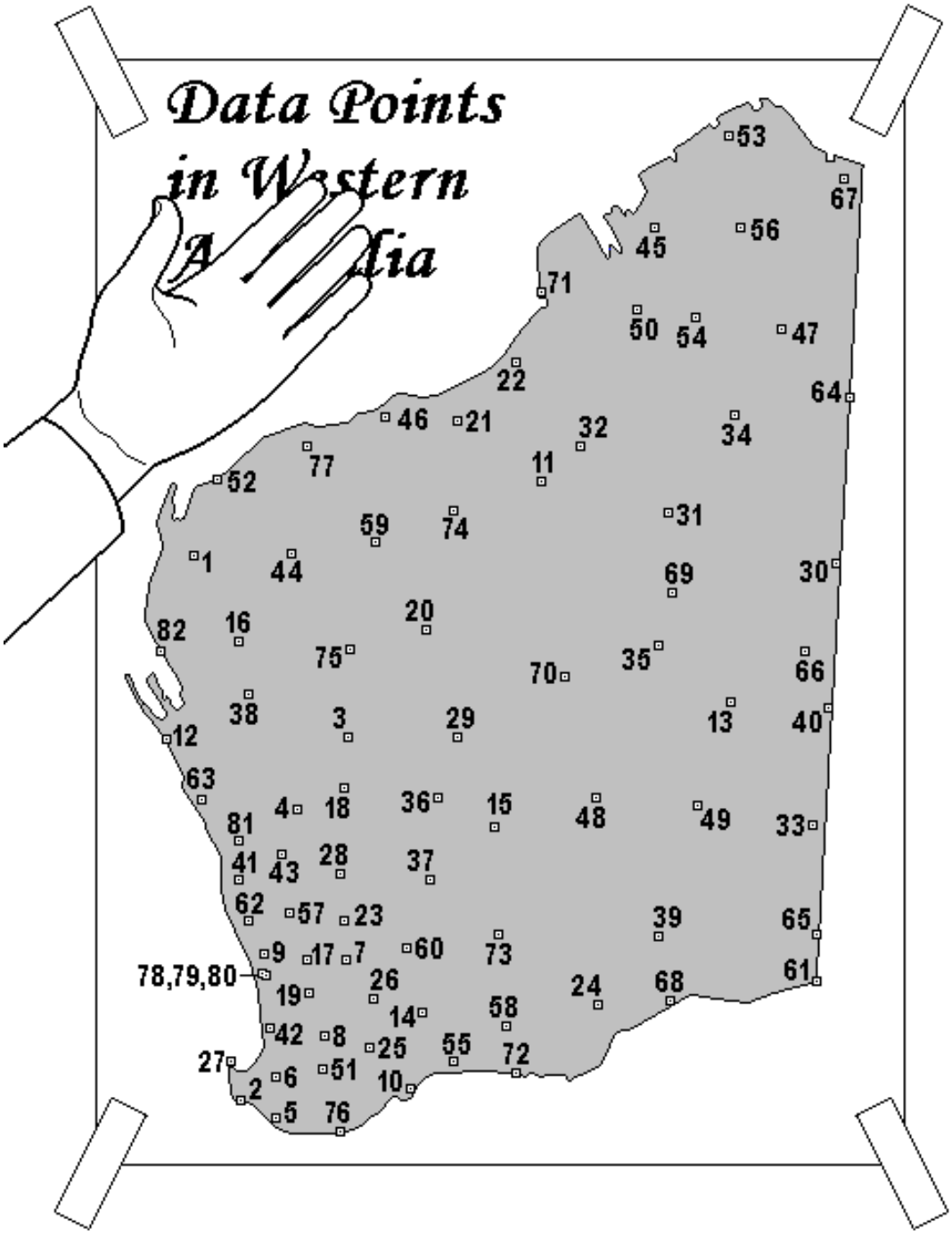
The choice of transition interval for LIVONN is up to the user. It is a matter of compromise between two objectives.

- A high value, say 0.6 or more, reduces the likelihood of peaks and troughs being captured among the computation points. Conversely, that likelihood can be boosted by a dense set of computation points (*ie* a small grid interval).

- A low value, say 0.2 or less, limits the number of intermediate values occurring among the computation points. It brings LIVONN closer to the discontinuities of nearest neighbour.

The transition interval illustrated in Figure 7-17 is one-third, which would mean, by (7-76), that α is also one-third.

Part Three: Case Studies



This chapter covers the derivation of datum transformations from Australian Geodetic Datum 1984 (AGD84) to the Geocentric Datum of Australia 1994 (GDA94). The area of application is Western Australia and there are 82 data points known in both datums. The coordinates are given in sub-appendix C.1.

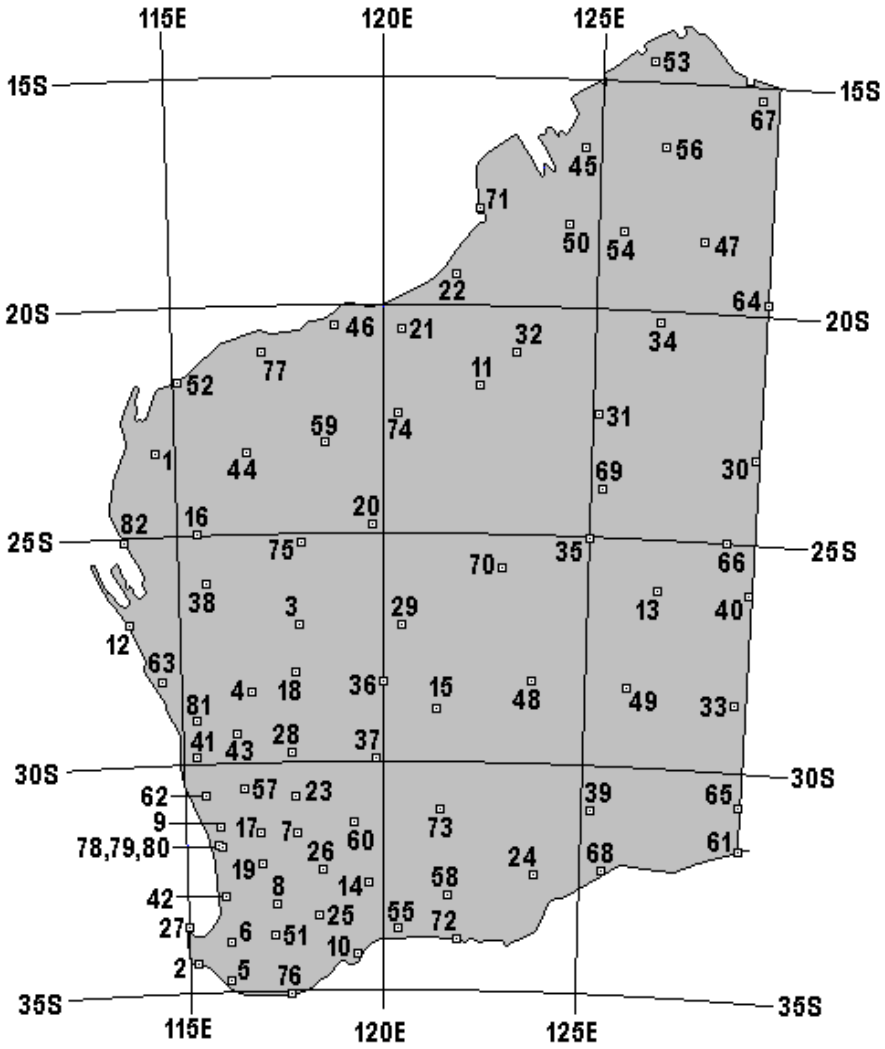


Figure 8-1: Data points for Western Australia.

The location of the data points is shown in Figure 8-1. Points 78, 79 and 80 are too close to be shown separately. Points 79 & 80 are horizontally 89 metres apart and are approximately 9.5km from point 78.

The points are the 82 stations of the STATEFIX GPS network, described in Agustan and Featherstone (2004) and more fully in Stewart *et al* (1998). The coordinates were computed in 1996 by the Geodesy Group at Curtin University of Technology in collaboration with the Western Australian Department of Land Administration (later superseded by Landgate).

The age of the data means that neither set of ellipsoidal heights could have come directly from GNSS. Enquiries made to Landgate and Curtin University failed to reveal the precise methods by which orthogonal heights were converted to ellipsoidal heights. In the case of AGD84 heights, it seems likely that the conversion used the 1971 astrogeodetic datum produced by the Division of National Mapping in Canberra. For the GDA94 heights, the geoid model used was probably an early NGA Earth Gravitational Model such as EGM96 or one its predecessors produced by Ohio State University.

The reason for leading the case studies with Western Australia is that the dataset is large enough to test the widest-possible range of methods, and small enough for the solution of linear equations to be easily manageable.

Western Australia was treated like the smaller datasets in one respect: all the data points were used as control points. In the case studies based on larger datasets, some of the data points were set aside for the purpose of an independent check on accuracy.

8.1 Application of basic methods other than MREs

Individual transformations were derived by Excel VBA subroutines specifically written for this study, each of them based on least-squares optimisation in one form or another, using the Western Australia dataset. Results quoted from published sources are also based on that dataset unless stated otherwise.

The 3-parameter conformal transformation obtained using the formulae in Section 4.3 consisted of the following: ΔX = -140.487m, ΔY = -33.935m and ΔZ=142.251m. The main statistics of the residuals are:

- Latitude RMS = 2.2920m. Longitude RMS = 1.7678m. Height RMS = 1.5223m.
- Horizontal RMS = 2.8946m. 3D RMS = 3.2705m.
- Mean Horizontal Distance = 2.6524m. Mean 3D Distance = 3.0518m.

NIMA (2000, page B.4-1) and NGA (2014, page D.4.1) give the "Australia and Tasmania" parameters for AGD84 to WGS84 as (-134±2)m, (-48±2)m and (149±2)m. The number of satellite stations used was 90. The difference in parameter values is probably due more to the area of coverage being wider than Western Australia than to differences between GDA94 and early realisations of WGS84.

Iliffe and Lott (2008, page 94) gives the AGD84→GDA94 parameters "for Australia" as -128.5m, -53.0m and 153.4m. The story behind these figures is given in the notes on Abridged Molodensky below.

The Standard Molodensky transformation obtained using the formulae in Section 4.4 consisted of the following: ΔX = -140.488m, ΔY = -33.932m and ΔZ=142.251m. The main statistics of the residuals are:

- Latitude RMS = 2.2917m. Longitude RMS = 1.7678m. Height RMS = 1.5223m.
- Horizontal RMS = 2.8943m. 3D RMS = 3.2703m.
- Mean Horizontal Distance = 2.6522m. Mean 3D Distance = 3.0517m

If the parameters from optimising the 3PC model are used in Standard Molodensky (which is probably common practice), the 3D RMS distance residual is also 3.2703m. There are slight differences in the other values; for example, the mean 3D Distance is 3.0519m.

The Abridged Molodensky transformation obtained using the formulae in Section 4.5 consisted of the following: ΔX = -140.479m, ΔY = -33.931m and ΔZ=142.243m. The main statistics of the residuals are:

- Latitude RMS = 2.2911m. Longitude RMS = 1.7664m. Height RMS = 1.5213m.
- Horizontal RMS = 2.8930m. 3D RMS = 3.2686m.
- Mean Horizontal Distance = 2.6506m. Mean 3D Distance = 3.0500m

If the parameters from optimising the 3PC model are used in Abridged Molodensky (which is probably common practice), the 3D RMS distance residual is also 3.2686m. There are slight differences in the other values; for example, the mean 3D Distance is 3.0516m.

ICSM (2014, Table 7-9) gives the parameters of the Abridged Molodensky transformation as ΔX = -128.5m, ΔY = -53.0m and ΔZ=153.4m. These were derived from 327 common points in Western Australia, South Australia and Queensland (the States which adopted AGD84). According to Featherstone *et al* (1999), the equations used were those for $\Delta\phi$ and $\Delta\lambda$, not Δh. (See the note on the 6-parameter variation of Abridged Molodensky.)

The Helmert Version 1 transformation obtained by the new four-stage method (HO1 to HO4 in Section 5.1) consisted of the following: ΔX=-115.838m, ΔY=-48.373m, ΔZ=144.760m, R_X=0.119712″, R_Y=0.383988″, R_Z=0.370396″, ΔS=3.689815ppm. The main statistics of the residuals are:

- Latitude RMS = 0.4452m. Longitude RMS = 0.5585m. Height RMS = 0.2548m.
- Horizontal RMS = 0.7142m. 3D RMS = 0.7583m.
- Mean Horizontal Distance = 0.6086m. Mean 3D Distance = 0.6653m.

The equivalent Helmert Version 2 transformation consisted of the following: ΔX=-115.838m, ΔY=-48.373m, ΔZ=144.760m, R_X=0.119711″, R_Y=0.383988″, R_Z=0.370396″, ΔS=3.689815ppm. The residuals are the same as for Helmert Version 1, so the main statistics are the same:

- Latitude RMS = 0.4452m. Longitude RMS = 0.5585m. Height RMS = 0.2548m.
- Horizontal RMS = 0.7142m. 3D RMS = 0.7583m.
- Mean Horizontal Distance = 0.6086m. Mean 3D Distance = 0.6653m.

The parameters for Helmert derived by the Procrustes method in Paláncz *et al* (2010) are -115.838m, -48.373m, 144.760m, -0.120″, -0.384″, -0.370″, 3.68981ppm. On the evidence of Paláncz *et al* (2008), the transformation was Helmert Version 2 and the rotation convention was CF (in contrast to PV which was adopted for this study); allowing for that, there is close agreement with the parameters derived in this study.

The Bursa-Wolf transformation (fully-linear version) obtained using the formulae in Section 4.6 consisted of the following: ΔX=-115.838m, ΔY=-48.373m, ΔZ=144.760m, R_X=0.119712″, R_Y=0.383990″, R_Z=0.370398″, ΔS=3.689812ppm. The main statistics of the residuals are:

- Latitude RMS = 0.4452m. Longitude RMS = 0.5585m. Height RMS = 0.2548m.
- Horizontal RMS = 0.7142m. 3D RMS = 0.7583m.
- Mean Horizontal Distance = 0.6086m. Mean 3D Distance = 0.6653m.

Featherstone (1997) quotes Bursa-Wolf parameters derived for AGD84 to WGS84 for Australia as a whole with a conspicuously different scale-change, namely 0.0983ppm. The distance analysis for this study confirmed that 3.689812ppm is correct for Western Australia.

The Molodensky-Badekas transformation (fully-linear version) obtained using the formulae in Section 4.9 consisted of the following: ΔX = -140.487m, ΔY = -33.935m, ΔZ = 142.251m, R_X = 0.119712″, R_Y = 0.383990″, R_Z = 0.370398″, ΔS = 3.689812ppm; of no less importance are the values X_m = 2862387.194m, Y_m = 4851873.712m, Z_m = -2887170.818m, obtained by averaging the AGD84 Cartesian coordinates. The main statistics of the residuals are:

- Latitude RMS = 0.4452m. Longitude RMS = 0.5585m. Height RMS = 0.2548m.
- Horizontal RMS = 0.7142m. 3D RMS = 0.7583m.
- Mean Horizontal Distance = 0.6086m. Mean 3D Distance = 0.6653m.

These statistics are identical with those for Bursa-Wolf. This supports the argument in Section 2.9 that the Bursa-Wolf and Molodensky-Badekas are equivalent.

The 8-parameter affine transformation obtained by the new EEH-based method (described in Section 5.3) was of type "SR" with a Version-1 rotation matrix applied before scaling. The midpoints used to derive the local level coordinates were (-2862387.19362, 4851873.71222, -2887170.81806) at 27.2936325115°S, 120.5386925915°E in AGD84, and (-2862527.68055, 4851839.77707, -2887028.56696) at 27.2923037131°S, 120.5400986365°E in GDA94. The parameters were as follows: $\Delta X'$=0m, $\Delta Y'$=0m, $\Delta Z'$=0m, $R_{X'}$=4.483580″, $R_{Y'}$=-4.043449″, $R_{Z'}$=2.390971″, ΔS_h=3.690667ppm and ΔS_v=3.314400ppm. The main statistics of the residuals are:

- Latitude RMS = 0.4449m. Longitude RMS = 0.5585m. Height RMS = 0.2550m.
- Horizontal RMS = 0.7140m. 3D RMS = 0.7582m.
- Mean Horizontal Distance = 0.6084m. Mean 3D Distance = 0.6651m.

The 9-parameter affine transformation obtained by the new four-stage method described in Section 5.4 was of type "SR" with a Version-1 rotation matrix applied before scaling. The parameters were as follows: ΔX=-112.169m, ΔY=-44.045m, ΔZ=144.312m, R_X=0.080927″, R_Y=0.436119″, R_Z=0.437119″, ΔS_X=4.1684ppm, ΔS_Y=3.1005ppm and ΔS_Z=3.4695ppm. The main statistics of the residuals are:

- Latitude RMS = 0.4209m. Longitude RMS = 0.5451m. Height RMS = 0.2523m.
- Horizontal RMS = 0.6887m. 3D RMS = 0.7334m.
- Mean Horizontal Distance = 0.5937m. Mean 3D Distance = 0.6506m.

Paláncz *et al* (2010) has parameters -112.169m, -44.047m, 144.311m $-0.081''$, $-0.436''$, $-0.437''$, 0.416838ppm, 0.310067ppm and 0.346938ppm obtained by the PZ method named after Paláncz and Zaletnyik (described earlier in Section 5.4). In that source, the parameters were near-identical to those obtained by "global minimisation" via a built-in function NMinimise in Mathematica (-112.169m, -44.046m, 144.312m $-0.081''$, $-0.436''$, $-0.437''$, 0.416842ppm, 0.310066ppm and 0.346943ppm). The signs of the rotations reflect the preference by Paláncz *et al* for the CF convention. The "error" quoted in each case was 6.642m, based on the square root of the sum of the squares of the distance residuals. Dividing by √82 gives a 3D RMS of 0.7335m.

The 12-parameter affine transformation obtained using the method described in Section 4.12 consisted of 3 translation parameters and a multiplying matrix. The translation parameters were ΔX=-110.101m, ΔY=-32.390m and ΔZ=159.563m. The matrix elements were

1 + 0.0000043358	-0.0000023499	0.0000022768
0.0000030106	1 + 0.0000017055	0.0000004165
-0.0000009402	-0.0000014281	1 + 0.0000045283

The main statistics of the residuals are:

- Latitude RMS = 0.4123m. Longitude RMS = 0.5439m. Height RMS = 0.2535m.
- Horizontal RMS = 0.6825m. 3D RMS = 0.7280m.
- Mean Horizontal Distance = 0.5848m. Mean 3D Distance = 0.6422m.

The new Standard Molodensky PCV transformation with 7 parameters obtained by the procedure described in Section 4.15 consisted of the following: ΔX_{hor}=-133.451m, ΔY_{hor}=-54.210m, $\Delta Z_{hor} = 153.254$m, R_Z=-0.151112″, ΔX_{ver}=-127.212m, ΔY_{ver}=-30.226m and ΔZ_{ver}=134.653m. The main statistics of the residuals are:

- Latitude RMS = 0.5052m. Longitude RMS = 0.5721m. Height RMS = 0.2760m.
- Horizontal RMS = 0.7632m. 3D RMS = 0.8116m.
- Mean Horizontal Distance = 0.6519m. Mean 3D Distance = 0.7149m.

Compared with Standard Molodensky, the 3D RMS distance residual is reduced by 75.18% and the horizontal RMS residual is reduced by 73.63%. This is despite the fact that the application of the 7-parameter variation involves no extra computation apart from the addition of R_Z to the longitude shift.

The new Standard Molodensky PCV transformation with 6 parameters obtained by the procedure described in Section 4.15 consisted of the following: ΔX_{hor}=-129.902m, ΔY_{hor}=-52.064m, $\Delta Z_{hor} = 153.218\text{m}$, ΔX_{ver}=-127.212m, ΔY_{ver}=-30.226m and $\Delta Z_{ver} = 134.653\text{m}$. The main statistics of the residuals are:

- Latitude RMS = 0.5388m. Longitude RMS = 0.5926m. Height RMS = 0.2760m.
- Horizontal RMS = 0.8009m. 3D RMS = 0.8472m.
- Mean Horizontal Distance = 0.6847m. Mean 3D Distance = 0.7428m.

Compared with Standard Molodensky, the 3D RMS distance residual is reduced by 74.09% and the horizontal RMS residual is reduced by 72.33%. This is despite the fact that the application of the 6-parameter variation involves no extra computation.

The new Abridged Molodensky PCV transformation with 7 parameters obtained by the procedure described in Section 4.16 consisted of the following: ΔX_{hor}=-133.445m, ΔY_{hor}=-54.198m, $\Delta Z_{hor} = 153.240\text{m}$, R_Z=-0.150949″, ΔX_{ver}=-127.212m, ΔY_{ver}=-30.226m and ΔZ_{ver}=134.653m. The main statistics of the residuals are:

- Latitude RMS = 0.5055m. Longitude RMS = 0.5713m. Height RMS = 0.2760m.
- Horizontal RMS = 0.7628m. 3D RMS = 0.8112m.
- Mean Horizontal Distance = 0.6519m. Mean 3D Distance = 0.7148m.

Compared with Abridged Molodensky, the 3D RMS distance residual is reduced by 75.18% and the horizontal RMS residual is reduced by 73.63%. This is despite the fact that the application of the 7-parameter partially-conformal variation involves no extra computation apart from the addition of R_Z to the longitude shift.

The new Abridged Molodensky PCV transformation with 6 parameters obtained by the procedure described in Section 4.16 consisted of the following: ΔX_{hor}=-129.899m, ΔY_{hor}=-52.054m, $\Delta Z_{hor} = 153.204\text{m}$, ΔX_{ver}=-127.212m, ΔY_{ver}=-30.226m and $\Delta Z_{ver} = 134.653\text{m}$. The main statistics of the residuals are:

- Latitude RMS = 0.5389m. Longitude RMS = 0.5919m. Height RMS = 0.2760m.
- Horizontal RMS = 0.8005m. 3D RMS = 0.8467m.
- Mean Horizontal Distance = 0.6844m. Mean 3D Distance = 0.7426m.

Compared with Abridged Molodensky, the 3D RMS distance residual is reduced by 74.10% and the horizontal RMS residual is reduced by 72.33%. This is despite the fact that the application of the 6-parameter variation involves no extra computation.

The values of ΔX_{hor}, ΔY_{hor} and ΔZ_{hor} are within a metre or two of the values of ΔX, ΔY and ΔZ given earlier for the Abridged Molodensky transformation in ICSM (2014). This bears out the final paragraph of Section 4.16. The ICSM shifts may be considered the values of ΔX, ΔY and ΔZ that give Abridged Molodensky the best horizontal fit over 3 States, but are not suitable for transforming h.

From these results, a few conclusions are offered:

- The 3-parameter transformations derived above have an inferior horizontal accuracy to ICSM's Abridged Molodensky transformation which "should only be used for low accuracy projects (accuracy no better than 5 m)" (ICSM [2014], page 35). This suggests that the RMS values are flattering, reflecting internal consistency.
- The 6-parameter variations on Standard and Abridged Molodensky reduce the 3D RMS residual by 74%, a significant benefit from separating ΔX_{ver}, ΔY_{ver} and ΔZ_{ver} from ΔX_{hor}, ΔY_{hor} and ΔZ_{hor}. Including R_Z as a 7th parameter in the Molodensky variations boosts that reduction to 75%.
- 7PC and its near-conformal versions improve on 3PC to the extent of reducing the 3D RMS residual by 77%.
- Of the affine transformations, the 9-parameter and 12-parameter version give modest improvements on 7PC. The 8-parameter affine transformation produces a relatively small difference between ΔS_h and ΔS_v and does not improve accuracy.

8.2 Application of multiple regression equations

The multiple regression equations considered were the fully-normalised MREs described in Section 2.17. That is to say that the intermediate coordinates U and V were defined in such a way that they varied between -1 and 1 over Western Australia.

The MREs described in this Section are listed in sub-appendix H.1. Only selected MREs are included in this Section.

The offset coordinates in degrees were -24.350 for latitude and 120.949 for longitude. The relative latitude and relative longitude were scaled as follows:

$$U = 0.09298\left(\phi_{\text{in deg}} + 24.350\right); \tag{8-1}$$

$$V = 0.12425\left(\lambda_{\text{in deg}} - 120.949\right). \tag{8-2}$$

The relationship between the geodetic and intermediate coordinates is illustrated in Figure 8-2.

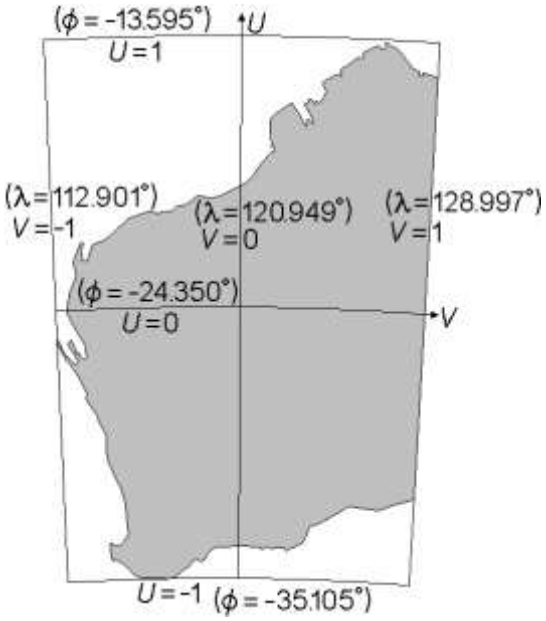


Figure 8-2: The relationship between (ϕ, λ) and (U, V).

The Western Australia dataset has 82 points and is small compared with the European datasets described in sub-appendices C.5 and C.6. It was decided to limit the MREs considered to those with no more than 40 terms. Taking account of Tables 2-4 to 2-8, this meant setting upper limits on the top power as follows:

- 5 for Ordinary MREs and Chebyshev MREs;
- 3 for North/South MREs, East/West MREs and Four-Quadrant MREs.

The recognised approach for deriving MREs is to apply least-squares optimisation on different combination of terms and to retain those terms which are statistically significant. The implementation of that approach for this study is the one described in Section 5.5: "eliminating ratios less than one" (ERLTO), where the term "ratio" is the absolute value of a parameter divided by its standard error (AP/SE). The full list of actual MREs is given in sub-appendix H.1.

The Ordinary MREs with top power 3 obtained by ERLTO had 13 terms for the latitude shift and (purely by coincidence) 13 for the longitude shift. The RMSs of the residuals are 0.2780m for latitude, 0.3029m for longitude and 0.4111m for horizontal distance.

The Ordinary MREs with top power 4 obtained by ERLTO had 18 terms for the latitude shift and (purely by coincidence) 18 for the longitude shift. The RMSs of the residuals are 0.2531m for latitude, 0.2539m for longitude and 0.3585m for horizontal distance.

The Ordinary MREs with top power 5 obtained by ERLTO had 28 terms for the latitude shift and (purely by coincidence) 28 for the longitude shift. The RMSs of the residuals are 0.2002m for latitude, 0.1994m for longitude and 0.2826m for horizontal distance.

The North/South MREs with top power 3 obtained by ERLTO had 18 terms for the latitude shift and 16 for the longitude shift. The RMSs of the residuals are 0.2570m for latitude, 0.2585m for longitude and 0.3645m for horizontal distance.

> Compared with Ordinary MREs with top power 4, the accuracy is 2.0% worse but was achieved with 2 fewer terms.

The East/West MREs with top power 3 obtained by ERLTO had 19 terms for the latitude shift and 15 for the longitude shift. The RMSs of the residuals are 0.2359m for latitude, 0.2869m for longitude and 0.3714m for horizontal distance.

> Compared with Ordinary MREs with top power 4, the accuracy is 3.6% worse but was achieved with 2 fewer terms. However, the accuracy of the latitude-shift MRE is 7.2% better despite only 1 additional term.

The Four-Quadrant MREs with top power 3 obtained by ERLTO had 19 terms for the latitude shift and 22 for the longitude shift. The RMSs of the residuals are 0.2152m for latitude, 0.2494m for longitude and 0.3294m for horizontal distance.

> Compared with Ordinary MREs with top power 4, the accuracy is 8.8% better but was achieved with 5 more terms. However, the accuracy of the latitude-shift MRE is 17.7% better despite only 1 additional term.

The Chebyshev MREs with top power 3 obtained by ERLTO had 13 terms for the latitude shift and 11 for the longitude shift. The RMSs of the residuals are 0.2788m for latitude, 0.3044m for longitude and 0.4128m for horizontal distance.

The Chebyshev MREs with top power 4 obtained by ERLTO had 18 terms for the latitude shift and 21 for the longitude shift. The RMSs of the residuals are 0.2528m for latitude, 0.2526m for longitude and 0.3574m for horizontal distance.

The Chebyshev MREs with top power 5 obtained by ERLTO had 32 terms for the latitude shift and 35 for the longitude shift. The RMSs of the residuals are 0.1988m for latitude, 0.1941m for longitude and 0.2779m for horizontal distance.

On the above evidence, a good compromise between accuracy and economy would be the Four-Quadrant latitude-shift MRE with top power 3 (19 terms) and the Ordinary longitude-shift MRE with top power 4 (18 terms). The RMSs of the residuals are 0.2152m for latitude, 0.2539m for longitude and 0.3328m for horizontal distance. The equations in this case are:

$$
\begin{aligned}
\Delta\phi('') = {} & 4.85208 + 0.27734U + 0.28524V - \\
& 0.06942U^2 - 0.30004UV - 0.02097V^3 + \\
& 0.22313UV^2 - 0.07558U^2V + 0.15936U^3V \\
& - 0.13175U^3V^2 \qquad \rightarrow \quad \text{if } U{>}0 \ \&\ V{>}0 \\
& \left.\begin{array}{l} - 45.19674U^2V^2 - 36.30299U^2V^3 + \\ 109.81505U^3V^2 + 60.82490U^3V^3 \end{array}\right\} \quad \text{if } U{>}0 \ \&\ V{\leq}0 \\
& - 0.27203U^2V^2 + 0.46726U^2V^3 \qquad \rightarrow \quad \text{if } U{\leq}0 \ \&\ V{>}0 \\
& + 1.32051U^2V^2 + 0.21603U^2V^3 + 0.94090U^3V^2 \qquad \rightarrow \quad \text{if } U{\leq}0 \ \&\ V{\leq}0
\end{aligned}
\tag{8-3}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 4.90452 - 0.50396U - 0.07512V + 0.14004U^2 + \\
& 0.06001UV - 0.09021V^2 + 0.22806U^3 + 0.05712U^4 - \\
& 0.45726U^3V + 0.06987UV^3 - 0.52469U^4V + \\
& 0.42484U^3V^2 - 0.53170U^2V^3 + 0.84934U^4V^2 - \\
& 0.28994U^3V^3 + 0.47094U^2V^4 + 0.52270U^4V^2 - \\
& 0.82683U^4V^4
\end{aligned}
\tag{8-4}
$$

The Chebyshev MREs in sub-appendix H.1 show a distinct tendency to have smaller coefficients than the MREs based on monomials. The significance of this can be deduced by comparing the initial ordinary MRE with top power 5 with the initial Chebyshev MRE with top 5; "initial" means "prior to application of ERLTO" so that they are different representations of the same polynomial. The L_1 norms of the coefficients were as follows:

- for the latitude shift, 40.8943 (Ord) and 18.7098 (Cheb);
- for the longitude shift, 64.6128 (Ord) and 23.2791 (Cheb).

Equivalence of the functions means that the lower Chebyshev L_1 norms are limits on the magnitudes of the ordinary MREs. (See subsections 2.17.1 and 2.17.5.) This implies that some of the seemingly-large monomial terms largely cancel each other out. It is logical to assume that this is true of the monomial-based MREs in general, including N/S, E/W and 4Q. This revelation is perhaps the most significant contribution of Chebyshev polynomials to the study of MREs in Western Australia.

Contour maps of the datum shifts generated by MREs are shown in sub-appendix I.1.1. Features of the maps include the following:

- There are marked changes in curvature, particularly for longitude shifts.
- Chebyshev MREs give similar contour maps to Ordinary MREs, reflecting the likelihood that although the discarded terms were not identical their contribution to the starting polynomials was small.
- Contour maps of Ordinary and Chebyshev MREs with top power 5 have irregularities, including $\Delta\phi$ contours that converge in the north-east.
- Contours of the Four-Quadrant MREs with top power 3 have irregularities on the western side including 90° bends, although that could be a reflection of distortion.

The MREs in this section produce RMSs of the residuals which are lower than all those in Section 8.1. In one sense this is not surprising, since the MRE pairings have at least 24 coefficients compared to the maximum of 12 parameters among the basic methods. The MRE pairing of (8-3) and (8-4) – which has a combined total of 37 terms – produces an RMS for horizontal distance just under half that of the 12-parameter affine transformation.

8.3 Application of composite methods

The composite methods applied in Western Australia were selected from those described in Chapter 7 and they included the use of gridded data (as opposed to just using the trend model with the surface-fitting technique). The methods for interpolating the gridded data were selected from those described in Chapter 6. The generic process used is illustrated in Figure 8-3.

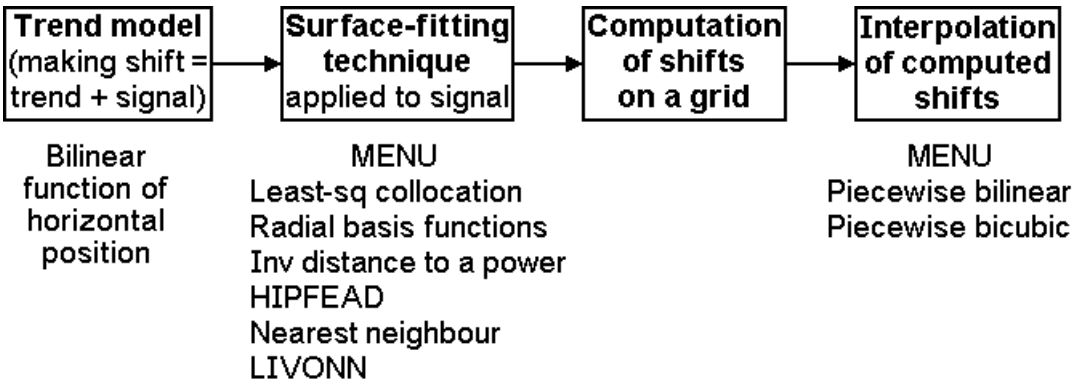


Figure 8-3: Composite methods chosen for AGD84→GDA94 in Western Australia.

The trend model chosen for the latitude shift (in arc-seconds) was

$$\Delta\phi_{BLM}('') = 4.82742 + 0.026276\left(\phi_{\text{in deg}} + 24.350\right) + 0.032961\left(\lambda_{\text{in deg}} - 120.949\right) - 0.0014730\left(\phi_{\text{in deg}} + 24.350\right)\left(\lambda_{\text{in deg}} - 120.949\right). \quad (8\text{-}5)$$

The trend model chosen for the longitude shift (in arc-seconds) was

$$\Delta\lambda_{BLM}('') = 4.89840 - 0.044790\left(\phi_{\text{in deg}} + 24.350\right) - 0.010518\left(\lambda_{\text{in deg}} - 120.949\right) + 0.0012290\left(\phi_{\text{in deg}} + 24.350\right)\left(\lambda_{\text{in deg}} - 120.949\right). \quad (8\text{-}6)$$

In each case, the trend model is the bilinear model that gives the least-squares fit to the shifts at the data points, hence the subscript BLM. This model is mathematically equivalent to the multiple regression equation with top power 1. The bilinear form has sufficient terms to ensure that

- the trend model includes the average shift as a constant term, and
- embodies the tilt which is evident from the contour maps I-1 to I-9 in Appendix I.

For both the latitude shift and the longitude shift, the signal component is that part which is unmodelled.

The surface-fitting methods tested for each signal were those shown in Figure 8-3. They are also listed below with the corresponding Sections.

- Least-squares collocation, with and without modification (7.1);
- Radial basis functions, with and without modification (7.6);
- Inverse distance to a power (7.5);
- Hybrid inverse power function embodying accelerated decline [HIPFEAD] (7.13);
- Nearest neighbour (7.7);
- Linear interpolation variation on nearest neighbour [LIVONN] (7.14).

Although – to varying degrees – these can be regarded as approximation methods in their own right, they are commonly used as methods to generate a regular grid. They were used for that purpose in this case study, although their merits as approximation methods were considered as part of their suitability for gridding. The grid was defined by intervals of 12′ (0.2°). There were 9095 grid points.

The 9095 grid points are different from the 259 points used to generate contour maps by LSS. Figure 8-4 shows the contours generated by LSS for the trend model defined by (8-5) and (8-6).

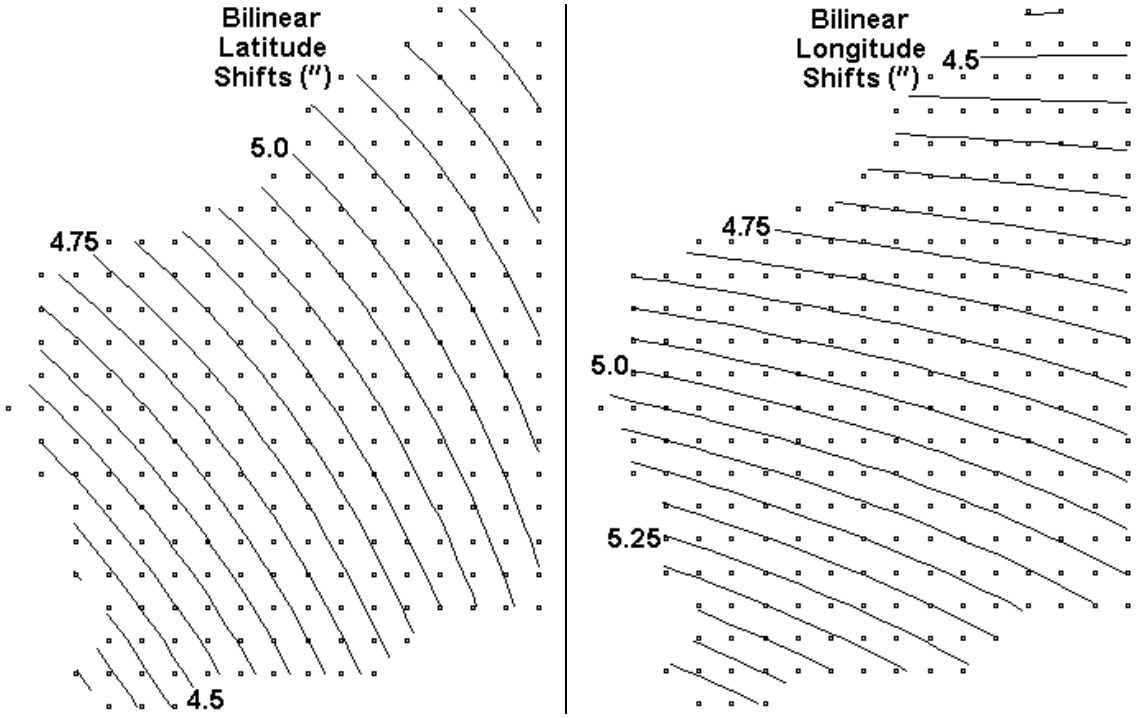


Figure 8-4: Contour maps of the trend models of AGD84→GDA94 in arc-seconds.

The methods used for interpolation of computed shifts were those shown in Figure 8-3. They are also listed below with the corresponding subsections.

- Piecewise bilinear interpolation (6.2);
- Piecewise bicubic interpolation (6.3, as in 6.3.3).

In both cases the interpolated shifts will differ from the original values at the data points. The differences only give a partial measure of accuracy as there are no independent check points.

The contour maps of datum arising from the composite methods are given in sub-appendix I.1.2. In the case of least-squares collocation and radial basis functions, only the modified versions are included. Observations about individual maps are made in this section in the context of the particular SFTs.

Least-squares collocation was applied with three covariance functions applied to the signals. These were Gaussian, SS20 and SS30, described in subsection 7.1. Variance (in square metres) was 0.34338 for the latitude signal and 0.75452 for the longitude signal. One way of setting the shape parameter is to define correlation length as median distance to nearest neighbour (MDNN) which is 137028m. Another way, applicable to SS20 and SS30 (both finite) is to define r_{max} as median distance to zero (MDZ), which is 188385m for latitude signal and 191513m for longitude signal. This led to six implementations:

- Gaussian with correlation length = 137028m, which is MDNN;
- Gaussian with correlation length = 68514m, which is half of MDNN;
- SS20 with r_{max} = 379395m, equivalent to setting correlation length to 137028m;
- SS20 with r_{max} = 188385m for latitude signal and r_{max} = 191513m for longitude signal;
- SS30 with r_{max} = 365690m, equivalent to setting correlation length to 137028m;
- SS30 with r_{max} = 188385m for latitude signal and r_{max} = 191513m for longitude signal.

All six implementations of LSC produced an unsatisfactory fit to the data-point signals. The problem was the signals' volatility in the south-west part of Western Australia, particularly near the data points 78-80 which are in close proximity. Both signals were changing by large amounts over short distances, with an element of oscillation.

To analyse the volatility, an area was selected with north-west corner 28°S, 116°E. It extended 5.8654° south and 5.0831° east. There were 21×27 nodes, approximately 25km apart. As the area was centred on the small Western Australian town Mukinbudin, it is described here as the Mukinbudin rectangle. There are 24 data points enclosed by the 12.5km boundary around the rectangle: the latitude signal (s_ϕ) varies from 0.6585m to -0.6706m and the longitude signal (s_λ) varies from 0.6239m to -0.4440m.

Table 8-1 shows that the implementations of least-squares collocation produce values of latitude signal and longitude signal which depart from these ranges by an unacceptable amount.

Table 8-1: Unmodified LSC: ranges of computed signal values at the nodes in the Mukinbudin rectangle

Covariance model	r_{max} (m) [ϕ]	max $\overline{s}_{\phi}$ (m)	min $\overline{s}_{\phi}$ (m)	r_{max} (m) [λ]	max $\overline{s}_{\lambda}$ (m)	min $\overline{s}_{\lambda}$ (m)
Gauss (L=137028m)	n/a	28.1315	-33.4922	n/a	20.9000	-17.8382
Gauss (L=68514m)	n/a	3.8514	-10.8116†	n/a	6.8155†	-2.4963
SS20	379395	2.2007	-3.9370	379395	2.3492	-1.5803
SS20	188358	2.3512	-5.7534††	191513	3.6302††	-1.5325
SS30	365690	2.2363	-3.4148†	365690	2.1512†	-1.4665
SS30	188385	2.5503	-3.7159†	191513	2.3053†	-1.6572

† at 32.2863°S, 116°E. †† at 32.5119°S, 116°E.

However, least-squares collocation was also applied with a modification devised for this study. It is precisely the one suggested in subsection 7.13.1 for overcoming volatility around clusters of data points, in this case the single cluster of data points 78, 79 and 80. Distance computations established these points were more than 45000m from the other 79 data points. A pseudo data point was created from data points 78, 79 and 80, using average coordinates and an average value of each signal. The modified least-squares collocation process was as follows:

- Least-squares collocation was applied to an interim dataset consisting of the pseudo data point and the data points outside the cluster.
- The residuals at the cluster points 78, 79 and 80 were computed (from the known values of the signals minus the values of the LSC approximation).
- A local limit-of-influence r_{max} was set to 45000m and a HISFEAD function was defined to interpolate the residuals at data points 78, 79 & 80.
- The modified LSC function was the original LSC function plus the HISFEAD function.

The effect of the modification on the surfacing fitting is considerable, as demonstrated in Table 8-2.

Table 8-2: Modified LSC: ranges of computed signal values at the nodes in the Mukinbudin rectangle

Covariance model	r_{max} (m) [ϕ]	max $\overline{s}_\phi$ (m)	min $\overline{s}_\phi$ (m)	r_{max} (m) [λ]	max $\overline{s}_\lambda$ (m)	min $\overline{s}_\lambda$ (m)
Gauss (L=137028m)	n/a	0.7858	-0.7624	n/a	0.8287	-0.4616
Gauss (L=68514m)	n/a	0.6838	-0.7237	n/a	0.6428	-0.4433
SS20	379395	0.7811	-0.7586	379395	0.8443	-0.4469
SS20	188358	0.6763	-0.7376	191513	0.6431	-0.4433
SS30	365690	0.7830	-0.7559	365690	0.8447	-0.4855
SS30	188385	0.6957	-0.7425	191513	0.6571	-0.4431

On this evidence, the lower values of the shape parameter (L or r_{max}) produce a narrower range of signal than the higher values.

For a wider check on the stability of least squares collocation, the datum-shift components $\overline{\Delta\phi}$ and $\overline{\Delta\lambda}$ were computed from trend + model at 9095 grid points.

Table 8-3: Comparison of Modified-LSC computed shifts with the known datum shifts

	Range of $\overline{\Delta\phi}$ over 9095 pts	Range of $\overline{\Delta\lambda}$ over 9095 pts
Modified LSC (Gauss, 137028m)	$4.1261'' \leq \overline{\Delta\phi} \leq 5.2412''$	$4.4274'' \leq \overline{\Delta\lambda} \leq 5.5971''$
Modified LSC (Gauss, 68514m)	$4.1260'' \leq \overline{\Delta\phi} \leq 5.2462''$	$4.4274'' \leq \overline{\Delta\lambda} \leq 5.5945''$
Modified LSC (SS20, 379395m, 379395m)	$4.1254'' \leq \overline{\Delta\phi} \leq 5.2417''$	$4.4274'' \leq \overline{\Delta\lambda} \leq 5.5997''$
Modified LSC (SS20, 188358m, 191513m)	$4.1260'' \leq \overline{\Delta\phi} \leq 5.2463''$	$4.4274'' \leq \overline{\Delta\lambda} \leq 5.5945''$
Modified LSC (SS30, 365690m, 365690m)	$4.1258'' \leq \overline{\Delta\phi} \leq 5.2415''$	$4.4274'' \leq \overline{\Delta\lambda} \leq 5.5967''$
Modified LSC (SS30, 188358m, 191513m)	$4.1260'' \leq \overline{\Delta\phi} \leq 5.2463''$	$4.4274'' \leq \overline{\Delta\lambda} \leq 5.5945''$
At the 82 data points, $4.2970'' \leq \Delta\phi \leq 5.1811''$ and $4.4979'' \leq \Delta\lambda \leq 5.4980''$.		

As Table 8-3 shows, the range of values was much the same for all 6 cases. Compared with the known shifts at the data points, the range of the latitude shifts increased by no more than 27.2% and the range of the longitude shifts increased by no more than 17.2%.

For the modified LSC shifts, the higher parameters for Gauss, SS20 and SS30 produced more regular contours than the lower parameters. (See Figures I-10 to I-15.) Interestingly, the higher parameters produced higher accuracy from gridding and interpolation as measured in Tables 8-10 and 8-11.

Radial basis functions were applied using all the examples described in subsection 7.6: inverse multiquadric (IMQ), multilog (ML), multiquadric (MQ), natural cubic spline (NCS) and thin plate spline (TPS). In each case, the shaping parameter R was set to 54347m, the value obtained from equation (7-50) which is the default used in Surfer.

All five RBFs produced an unsatisfactory fit to the data-point signals. As with least-squares collocation, the problem was volatility in the south-west part of Western Australia, particularly near the data points 78-80 which are in close proximity. The Mukinbudin rectangle used to analyse least-squares collocation was used again for the RBFs. Table 8-4 shows that each of them produces values of latitude signal and longitude signal which depart from these ranges by an unacceptable amount.

Table 8-4: Unmodified RBF: ranges of computed signal values at the nodes in the Mukinbudin rectangle

Radial Basis Function	R (m)	max $\overline{s}_\phi$ (m)	min $\overline{s}_\phi$ (m)	R (m)	max $\overline{s}_\lambda$ (m)	min $\overline{s}_\lambda$ (m)
Inverse multiquadric (IMQ)	54347	3.0972	-3.4831†	54347	2.1903†	-2.0082
Multilog (ML)	54347	3.1655	-4.7289†	54347	2.9770†	-2.0515
Multiquadric (MQ)	54347	3.2017	-6.0751†	54347	3.8283†	-2.0748
Natural cubic spline (NCS)	54347	3.3324	-8.6223†	54347	5.4329†	-2.1013
Thin plate spline (TPS)	54347	3.2253	-7.4081†	54347	4.6690†	-2.1016

† at 32.2863°S, 116°E

However, the radial basis functions were also applied with the same modification used for least-squares collocation. The data points 78, 79 and 80 form a cluster, each of them more than 45000m from the other 79 data points. A pseudo data point was created from data points 78, 79 & 80, using average coordinates and an average value of each signal. The modified RBF interpolation process was as follows:

- The radial basis function was applied to an interim dataset consisting of the pseudo data point and the data points outside the cluster.
- The residuals at the cluster points 78, 79 and 80 were computed (from the known values of the signals minus the values of the RBF approximation).
- A local limit-of-influence r_{max} was set to 45000m and a HISFEAD function was defined to interpolate the residuals at data points 78, 79 & 80.
- The modified radial basis function was the original RBF function plus the HISFEAD function.

The effect of the modification on the surfacing fitting is considerable, as demonstrated in Table 8-5.

Table 8-5: Modified RBF: ranges of computed signal values at the nodes in the Mukinbudin rectangle

Radial Basis Function	R (m)	max $\overline{s}_\phi$ (m)	min $\overline{s}_\phi$ (m)	R (m)	max $\overline{s}_\lambda$ (m)	min $\overline{s}_\lambda$ (m)
Inverse multiquadric	54347	0.6920	-0.6716	54347	0.6479	-0.4476
Multilog (ML)	54347	0.7384	-0.7234	54347	0.7231	-0.4490
Multiquadric (MQ)	54347	0.7619	-0.7473	54347	0.7605	-0.4569
Natural cubic spline	54347	1.1209	-0.5426	54347	0.7725	-0.4572
Thin plate spline (TPS)	54347	1.1546	-0.6207	54347	0.7132	-0.4453

In this instance, the inverse multiquadric produces a narrower range of signal than the other RBFs.

For a wider check on the stability of radial basis functions, the datum-shift components $\overline{\Delta\phi}$ and $\overline{\Delta\lambda}$ were computed from trend + model at 9095 grid points.

Table 8-6 shows significant variations in the range of values for the different types of RBF. Compared with the known shifts at the data points, the increase in the range of the latitude shifts is 25.4% for IMQ, 24.0% for ML, 22.7% for MQ, 34.8% for NCS and 26.3% for TPS; the increase in the range of the longitude shifts is 16.7% for IMQ, 15.8% for ML, 14.3% for MQ, 20.6% for NCS and 20.0% for TPS. On this evidence, ML is the least volatile; NCS is the most volatile, followed by TPS.

Table 8-6: Comparison of Modified-RBF computed shifts with the known datum shifts

	Range of $\overline{\Delta\phi}$ over 9095 points	Range of $\overline{\Delta\lambda}$ over 9095 points
Inverse multiquadric (R=54347m)	$4.1269'' \le \overline{\Delta\phi} \le 5.2360''$	$4.4291'' \le \overline{\Delta\lambda} \le 5.5958''$
Multilog (R=54347m)	$4.1266'' \le \overline{\Delta\phi} \le 5.2231''$	$4.4393'' \le \overline{\Delta\lambda} \le 5.5974''$
Multiquadric (R=54347m)	$4.1285'' \le \overline{\Delta\phi} \le 5.2136''$	$4.4604'' \le \overline{\Delta\lambda} \le 5.6032''$
Natural cubic spline (R=54347m)	$4.1242'' \le \overline{\Delta\phi} \le 5.3160''$	$4.4411'' \le \overline{\Delta\lambda} \le 5.6476''$
Thin plate spline (R=54347m)	$4.1279'' \le \overline{\Delta\phi} \le 5.2445''$	$4.4178'' \le \overline{\Delta\lambda} \le 5.6176''$
At the 82 data points, $4.2970'' \le \Delta\phi \le 5.1811''$ and $4.4979'' \le \Delta\lambda \le 5.4980''$.		

For the modified RBF latitude shifts, the contour maps are fairly regular in terms of the shape and spacing of contours. For the modified RBF longitude shifts, this can only be said of the cases where the RBF was NCS or TPS. (See Figures I-16 to I-20.)

Inverse distance to a power, described in Section 7.5, was the first of the strictly-bounded SFTs to be applied. It was applied with each of the powers 1, 2, 3 and 4. As this is a strictly bounded form of interpolation, there is no need for the kind of analysis given in Tables 8-1 to 8-6.

Inverse distance to the power 1 produced the most regular contours. This characteristic is deceptive, because this form of interpolation is not smooth. This is demonstrated in Table 8-7, which shows the longitude signal at and around data point 53. Over the first kilometre from that point the signal drops 0.14495m on average; over the second kilometre it drops 0.12744.

Table 8-7: Longitude signal in metres around data point 53, as computed from inverse distance to the power 1

	126.11648°	126.12576°	126.13504°	126.14432°	126.15360°
-14.55137°	1.94995	2.01627	2.04401	2.01618	1.94977
-14.56041°	2.01602	2.11636	2.17097	2.11628	2.01586
-14.56945°	2.04356	2.17087	2.31587	2.17079	2.04340
-14.57849°	2.01546	2.11597	2.17069	2.11589	2.01530
-14.58753°	1.94863	2.01513	2.04296	2.01505	1.94844

This slightly concave variation of a conical peak is illustrated in Figure 8-5.

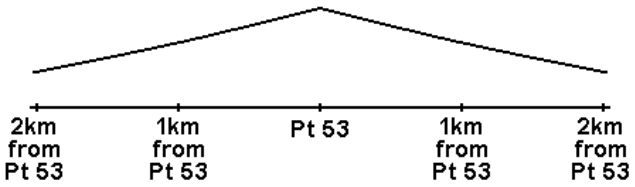


Figure 8-5: Longitude signal along any profile through data point 53, as computed from inverse distance to the power 1.

Inverse distance to a power produced increasingly irregular contours as the power rose from 1 to 4. Although irregular contours are not necessarily an indication of a bad fit, they were (in places) disturbingly jagged for the longitude shifts in the case where the power was 4. (See Figures I-21 to I-24.)

The next form of surface fitting to be applied was HIPFEAD as defined in Section 7.13. As this is a strictly bounded form of interpolation, there is no need for the kind of analysis given in Tables 8-1 to 8-6. The two versions of HIPFEAD are HISFEAD and HICFEAD, described in subsections 7.13.1 and 7.13.2 respectively.

In Section 7.13, it was noted that both HISFEAD and HICFEAD require r_{max} to be sufficiently large to ensure that *every* possible point of interest is within r_{max} of at least one data point. The point 23.977332°S 121.768162°E is just over 223716m from data points 20, 70 and 74. Some points in the polygon formed by data points 11, 31, 69, 35 and 70 could be slightly further from the nearest data point. As a result, the smallest of the values of the limit-of-influence r_{max} was set at 240000m. The others were 360000m and 480000m.

The limit-of-influence had a big influence on the number of data points within range of the computation (grid) points:

- For r_{max} = 240000m, 9 grid points had 1 data point within range, 45 had 2, 82 had 3 and 123 had 4 or more.
- For r_{max} = 360000m, 6 grid points had 3 data points within range and 253 had 4 or more.
- For r_{max} = 480000m, all 259 grid points had 4 or more data points within range,

For the HISFEAD contour maps, the higher the limit-of-influence, the greater the resemblance to the contour maps for inverse distance to the power 2. This is very much to be expected, because the inverse-square part of HISFEAD will cover a greater area. For the longitude shifts, the smallest value of r_{max} gave the least regular contour maps, although that could simply mean it is better than the other limits for representing distortion. (See Figures I-25 to I-27.)

For the HICFEAD contour maps, the higher the limit-of-influence, the greater the resemblance to the contour maps for inverse distance to the power 3. This is very much to be expected, because the inverse-square part of HISFEAD will cover a greater area. For the longitude shifts, the smallest value of r_{max} gave the least regular contour maps, although that could simply mean it is better than the other limits for representing distortion. (See Figures I-28 to I-30.)

The next form of surface fitting to be applied was nearest neighbour as described in Section 7.7. As this is a strictly bounded form of interpolation, there is no need for the kind of analysis given in Tables 8-1 to 8-6. Several of the datum-shift contours are jagged, particularly for the longitude. (See Figure I-31.)

Table 8-8 shows the latitude signal over an area of 13×13 nodes, approximately 25km apart. The nodes taking the signal value from the same data point are partitioned with bold boundary lines. The constant property within each partitioned area will not be replicated in the datum shift because the signal is added to the trend model.

Table 8-8: Latitude signal for part of the Mukinbudin rectangle using nearest neighbour

ϕ▼λ▶	116.00	116.25	116.51	116.76	117.02	117.27	117.52	117.78	118.03	118.29	118.54	118.80	119.05
-28.00	-0.289	-0.289	-0.289	-0.289	-0.289	-0.053	-0.053	-0.053	-0.053	-0.053	-0.053	-0.053	0.671
-28.23	-0.289	-0.289	-0.289	-0.289	-0.289	-0.289	-0.053	-0.053	-0.053	-0.053	-0.053	-0.053	0.671
-28.45	-0.289	-0.289	-0.289	-0.289	-0.289	-0.289	-0.053	-0.053	-0.053	-0.053	-0.053	-0.053	0.671
-28.68	-0.289	-0.289	-0.289	-0.289	-0.289	-0.289	-0.053	-0.053	-0.053	-0.053	-0.053	-0.053	0.671
-28.90	-0.077	-0.077	-0.077	-0.289	-0.289	-0.289	-0.289	-0.053	-0.053	-0.053	-0.053	-0.053	0.671
-29.13	-0.077	-0.077	-0.077	-0.077	-0.077	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.087
-29.35	-0.077	-0.077	-0.077	-0.077	-0.077	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.087
-29.58	-0.077	-0.077	-0.077	-0.077	-0.077	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.087
-29.80	-0.077	-0.077	-0.077	-0.077	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.087
-30.03	0.106	0.070	0.070	0.070	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.098	-0.087
-30.26	0.070	0.070	0.070	0.070	0.070	-0.098	-0.098	-0.098	-0.141	-0.141	-0.141	-0.141	-0.087
-30.48	0.070	0.070	0.070	0.070	0.070	-0.141	-0.141	-0.141	-0.141	-0.141	-0.141	-0.141	-0.177
-30.71	0.161	0.070	0.070	0.070	0.070	-0.141	-0.141	-0.141	-0.141	-0.141	-0.141	-0.177	-0.177

The next form of surface fitting to be applied was LIVONN as defined in Section 7.14. As this is a strictly bounded form of interpolation, there is no need for the kind of analysis given in Tables 8-1 to 8-6. The chosen transition intervals were 33.3333% (which matches the one illustrated in Figure 7-17) and 50%.

Table 8-9 shows a LIVONN latitude signal at the same nodes as Table 8-8. The values from neighbouring data points are still present (within bold boundaries) but occur in smaller areas. The shaded cells contain interpolated values and show plausible transitions between the "flat" area. The possibility of a discontinuity as per Figure 7-18 cannot be ruled out, but there is a pattern of continuity which is not present in Table 8-8.

Table 8-9: Latitude signal for part of the Mukinbudin rectangle using LIVONN with transition interval 33.333%.

ϕ▼λ▶	116.00	116.25	116.51	116.76	117.02	117.27	117.52	117.78	118.03	118.29	118.54	118.80	119.05
-28.00	-0.209	-0.257	-0.275	-0.265	-0.222	-0.127	-0.053	-0.053	-0.053	-0.062	-0.088	0.203	0.425
-28.23	-0.203	-0.276	-0.289	-0.289	-0.268	-0.171	-0.072	-0.053	-0.053	-0.056	0.050	0.224	0.448
-28.45	-0.178	-0.269	-0.289	-0.289	-0.289	-0.215	-0.118	-0.074	-0.064	-0.069	0.087	0.253	0.456
-28.68	-0.121	-0.232	-0.272	-0.289	-0.279	-0.230	-0.159	-0.113	-0.061	-0.063	-0.064	0.281	0.451
-28.90	-0.019	-0.145	-0.175	-0.213	-0.226	-0.225	-0.200	-0.075	-0.074	-0.073	-0.073	0.302	0.354
-29.13	-0.043	-0.077	-0.082	-0.119	-0.155	-0.088	-0.147	-0.089	-0.087	-0.085	-0.083	-0.092	0.203
-29.35	-0.061	-0.077	-0.077	-0.083	-0.083	-0.091	-0.096	-0.097	-0.096	-0.093	-0.095	-0.092	-0.090
-29.58	-0.038	-0.077	-0.077	-0.079	-0.086	-0.095	-0.098	-0.098	-0.098	-0.101	-0.095	-0.092	-0.089
-29.80	0.007	-0.050	-0.047	-0.036	-0.090	-0.096	-0.098	-0.098	-0.098	-0.102	-0.107	-0.092	-0.089
-30.03	0.091	0.013	0.020	0.022	-0.021	-0.072	-0.099	-0.098	-0.103	-0.108	-0.112	-0.092	-0.089
-30.26	0.109	0.081	0.069	0.062	0.024	-0.116	-0.116	-0.118	-0.120	-0.121	-0.121	-0.121	-0.118
-30.48	0.113	0.071	0.070	0.070	0.025	-0.069	-0.136	-0.139	-0.137	-0.133	-0.129	-0.158	-0.141
-30.71	0.121	0.075	0.070	0.070	0.012	-0.090	-0.141	-0.141	-0.141	-0.138	-0.152	-0.163	-0.170

In terms of shape and spacing, LIVONN with a transition interval of 33.3333% produced more regular contours for the latitude and longitude datum shifts than nearest neighbour. LIVONN with a transition interval of 50% had a similar advantage over nearest neighbour, except for

the longitude shift near the south coast between 122°E and 124°E. (See Figures I-32 and I-33.)

For surface-fitting techniques as a whole, accuracy comparisons are difficult in the absence of validation points. (The view adopted for this study was that the Australian dataset was insufficiently large to set some points aside.) There is, however, a partial basis for accuracy comparisons when the "trend + signal" model is used as a gridding method.

When gridding is used, the final computed shifts are computed by bilinear interpolation (BL) or bicubic interpolation (BC). The values of $\overline{\Delta\phi}$ and $\overline{\Delta\lambda}$ at the data points will not necessarily be the same as the original data values. The horizontal RMS residual for the data points will provide an indication of accuracy, albeit one that should be used with caution.

The gridding was applied at 9095 points, spaced 12′ apart (the same points as those referred to in Tables 8-3 and 8-6). Table 8-10 gives the horizontal RMS residual over the 82 data points when bilinear interpolation was used. Table 8-11 does so when bicubic interpolation was used. In both tables, the methods are listed in ascending order of RMS.

Table 8-10: Accuracy comparison for gridding with bilinear (BL) interpolation in Western Australia, based on residuals at the data points

Surface-Fitting Technique (with parameters)	Horizontal RMS Residual For Data Points (m)
Nearest Neighbour (no parameters)	0.00187 (beating 0.00197 for BC)
LIVONN (33.3333)	0.00202 (beating 0.00243 for BC)
Inverse Distance to a Power (4)	0.00204 (beating 0.00210 for BC)
LIVONN (50)	0.00204 (beating 0.00296 for BC)
HICFEAD (240000, 240000)	0.00225 (beating 0.00229 for BC)
HICFEAD (360000, 360000)	0.00236
HICFEAD (480000, 480000)	0.00243
Inverse Distance to a Power (3)	0.00259
Least-Squares Collocation (Gauss, 137028, 137028)	0.00565
Least-Squares Collocation (SS20, 379397, 379395)	0.00637
Least-Squares Collocation (SS30, 365690, 365690)	0.00654
HISFEAD (240K, 240K)	0.00728
Radial Basis Function (MQ, 54347)	0.00739
Radial Basis Function (ML, 54347)	0.01051
HISFEAD (360K, 360K)	0.01092
HISFEAD (480K, 480K)	0.01378
Radial Basis Function (IMQ, 54347)	0.02388
Least-Squares Collocation (Gauss, 68514, 68514)	0.02522
Least-Squares Collocation (SS30, 188385, 191513)	0.02615
Least-Squares Collocation (SS20, 188385, 191513)	0.02666
Inverse Distance to a Power (2)	0.02835
Inverse Distance to a Power (1)	0.53587
Radial Basis Function (TPS, 54347)	0.78844 (beating 0.78847 for BC)
Radial Basis Function (NCS, 54347)	1.23413 (beating 1.23424 for BC)

Table 8-11: Accuracy comparison for gridding with bicubic (BC) interpolation in Western Australia, based on residuals at the data points

SFT (with parameters) Applied With Bicubic Interpolation	Horizontal RMS Residual For Data Points (m)
Nearest Neighbour (no parameters)	0.00197
Inverse Distance to a Power (4)	0.00210
HICFEAD (240000, 240000)	0.00229
HICFEAD (360000, 360000)	0.00235 (beating 0.00236 for BL)
HICFEAD (480000, 480000)	0.00237 (beating 0.00243 for BL)
Inverse Distance to a Power (3)	0.00243 (beating 0.00259 for BL)
LIVONN (33.3333)	0.00243
LIVONN (50)	0.00296
Least-Squares Collocation (Gauss, 137028, 137028)	0.00389 (beating 0.00565 for BL)
Least-Squares Collocation (SS30, 365690, 365690)	0.00434 (beating 0.00654 for BL)
HISFEAD (240K, 240K)	0.00438 (beating 0.00728 for BL)
Least-Squares Collocation (SS20, 379397, 379395)	0.00439 (beating 0.00637 for BL)
Radial Basis Function (MQ, 54347)	0.00559 (beating 0.00739 for BL)
Radial Basis Function (ML, 54347)	0.00615 (beating 0.01051 for BL)
HISFEAD (360K, 360K)	0.00644 (beating 0.01092 for BL)
HISFEAD (480K, 480K)	0.00809 (beating 0.01378 for BL)
Least-Squares Collocation (Gauss, 68514, 68514)	0.01420 (beating 0.02522 for BL)
Radial Basis Function (IMQ, 54347)	0.01423 (beating 0.02388 for BL)
Least-Squares Collocation (SS30, 188385, 191513)	0.01525 (beating 0.02615 for BL)
Least-Squares Collocation (SS20, 188385, 191513)	0.01540 (beating 0.02666 for BL)
Inverse Distance to a Power (2)	0.01618 (beating 0.02835 for BL)
Inverse Distance to a Power (1)	0.50156 (beating 0.53587 for BL)
Radial Basis Function (TPS, 54347)	0.78847
Radial Basis Function (NCS, 54347)	1.23424

Some conclusions can be drawn.

- All but the last 3 methods in Tables 8-10 and Table 8-11 have much smaller data-point residuals that the MREs in equations (8-3) and (8-4).
- The last 3 methods in the tables can be considered totally unsuitable for transforming AGD84 to GDA94, since they have higher data-point residuals than the MREs. This was perhaps unexpected for radial basis functions natural cubic spline (NCS) and thin plate spline (TPS). In both cases the residuals in the latitude shifts are largest on the eastern side. (It was noted earlier that these were the most volatile RBFs on the evidence of the range of computed values over the 9095 grid points.)
- The data-point residuals for nearest neighbour and LIVONN are inevitably affected by the preservation of data-point signals via flat areas. (Recall Figure 7-10 and Figure 7-17.) The position of those methods in the above tables is therefore flattering. The results for inverse distance to the power 4 (and possibly 3) may also be flattering because of the relative flatness around the data points illustrated in Figure 7-9.
- For most methods, bicubic interpolation is more accurate than bilinear interpolation.
- For a more satisfactory accuracy check, there needs to be enough data points for some to be set aside for validation. This would remove the bias in accuracy comparisons towards SFTs that have flatness or relative flatness around the control points.

This chapter covers the derivation of datum transformations from Ordnance Survey Great Britain 1936 (OSGB) to WGS84. The area of application is Great Britain and there are 44 data points known in both datums. Given that the data was provided by E. J. Price who received it circa 2001, the "WGS84" designation should be regarded as an early realisation of the datum. The coordinates are given in sub-appendix C.2.

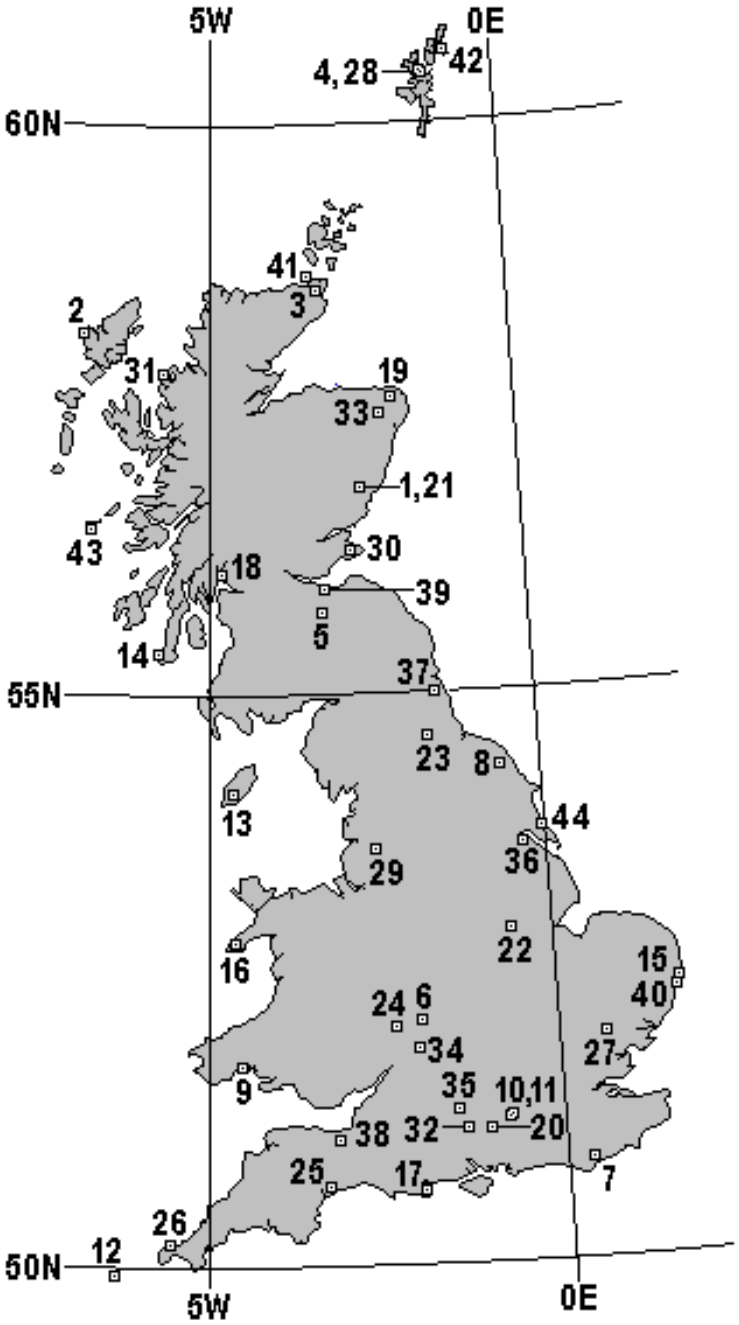


Figure 9-1: Data points for Great Britain (3D).

The location of the data points is shown in Figure 9-1. They are numbered 1 to 44 here, but are cross-referenced to official point identifications in sub-appendix C.2. The closest data

points are 1 and 21 which are horizontally 531m apart. Other pairs separated by short distances are 10 & 11 (1838m), 4 & 28 (5403m) and 15 & 40 (10079m).

Advice from the UK Defence Geographic Centre was that the points are Doppler stations and that the WGS coordinates came from Doppler precise positioning, not GPS. Geoid models must have been used to convert orthometric heights to ellipsoidal heights. (It was not possible to establish which of the many possibilities those models were.) This suggests that the ellipsoidal heights were much less accurate than the latitudes and longitudes.

Out of the thousands of points there are in Great Britain, these 44 points were the only ones where (ϕ, λ, h) values in two datums were available for this study. As the dataset is one of the smaller ones acquired for this study, this case study is limited to transformations with a small number of parameters, thus ensuring plenty of degrees of freedom. All the data points were used as control points.

Individual transformations were derived by Excel VBA subroutines specifically written for this study, each of them based on least-squares optimisation in one form or another.

The 3-parameter conformal transformation obtained using the formulae in Section 4.3 consisted of the following: ΔX = 376.414m, ΔY = -111.300m and ΔZ = 431.653m. The main statistics of the residuals are:

- Latitude RMS = 7.5288m. Longitude RMS = 2.7478m. Height RMS = 1.5963m.
- Horizontal RMS = 8.0146m. 3D RMS = 8.1720m.
- Mean Horizontal Distance = 7.4209m. Mean 3D Distance = 7.6274m.

NIMA (2000, page B.5-4) and NGA (2014, page D.5-5) give the "Mean Solution" parameters for OSGB36 to WGS84 as (375±10)m, (-111±10)m and (431±15)m. The number of satellite stations used was 38.

The Standard Molodensky transformation obtained using the formulae in Section 4.4 consisted of the following: ΔX = 376.414m, ΔY = -111.291m and ΔZ = 431.600m. The main statistics of the residuals are:

- Latitude RMS = 7.5257m. Longitude RMS = 2.7466m. Height RMS = 1.5964m.
- Horizontal RMS = 8.0112m. 3D RMS = 8.1687m.
- Mean Horizontal Distance = 7.4178m. Mean 3D Distance = 7.6244m.

If the parameters from optimising the 3PC model are used in Standard Molodensky (which is probably common practice), the 3D RMS distance residual is also 8.1687m, but there are slight differences in the other values.

The Abridged Molodensky transformation obtained using the formulae in Section 4.5 consisted of the following: ΔX = 376.318m, ΔY = -111.284m and ΔZ = 431.656m. The main statistics of the residuals are:

- Latitude RMS = 7.5079m. Longitude RMS = 2.7497m. Height RMS = 1.5961m.
- Horizontal RMS = 7.9956m. 3D RMS = 8.1534m.
- Mean Horizontal Distance = 7.4036m. Mean 3D Distance = 7.6104m.

If the parameters from optimising the 3PC model are used in Abridged Molodensky (which is probably common practice), the 3D RMS distance residual is 8.1539m, which is fractionally higher.

The Helmert Version 1 transformation obtained by the new four-stage method (HO1 to HO4 in Section 5.1) consisted of the following: ΔX = 445.181m, ΔY = -161.834m, ΔZ = 542.616m, R_X = -0.732442″, R_Y = 0.279006″, R_Z = 1.607763″, ΔS = -20.686291ppm. The main statistics of the residuals are:

- Latitude RMS = 1.5988m. Longitude RMS = 1.5863m. Height RMS = 1.1298m.
- Horizontal RMS = 2.2522m. 3D RMS = 2.5196m.
- Mean Horizontal Distance = 1.9452m. Mean 3D Distance = 2.2691m.

The equivalent Helmert Version 2 transformation consisted of the following: ΔX = 445.181m, ΔY = -161.834m, ΔZ = 542.616m, R_X = -0.732444″, R_Y = 0.279000″, R_Z = 1.607764″, ΔS = -20.686291ppm. The residuals are the same as for Helmert Version 1, so the main statistics are the same:

- Latitude RMS = 1.5988m. Longitude RMS = 1.5863m. Height RMS = 1.1298m.
- Horizontal RMS = 2.2522m. 3D RMS = 2.5196m.
- Mean Horizontal Distance = 1.9452m. Mean 3D Distance = 2.2691m.

The Bursa-Wolf transformation (fully-linear version) obtained using the formulae in Section 4.6 consisted of the following: ΔX = 445.181m, ΔY = -161.834m, ΔZ = 542.616m, R_X

= -0.732432″, R_Y = 0.278998″, R_Z = 1.607732″, ΔS = -20.686319ppm. The main statistics of the residuals match those of Helmert Versions 1 and 2:

- Latitude RMS = 1.5988m. Longitude RMS = 1.5863m. Height RMS = 1.1298m.
- Horizontal RMS = 2.2522m. 3D RMS = 2.5196m.
- Mean Horizontal Distance = 1.9452m. Mean 3D Distance = 2.2691m.

The parameters listed in NGA (2008) are listed as 446.448m, -125.157, 542.06, 0.150″, 0.2470″, 0.8421″, -20.49ppm. ESRI (2012) has the same figures except for the scale change (-20.489ppm).

The Molodensky-Badekas transformation (fully-linear version) obtained using the formulae in Section 4.9 consisted of the following: ΔX = 376.414m, ΔY = -111.300m, ΔZ = 431.635m, R_X = -0.732432″, R_Y = 0.278998″, R_Z = 1.607732″, ΔS = -20.686319ppm; of no less importance are the values X_m = 370212.608m, Y_m = -157444.673m, Z_m = 5147839.809m, obtained by averaging the OSGB36 Cartesian coordinates. The main statistics of the residuals are:

- Latitude RMS = 1.5988m. Longitude RMS = 1.5863m. Height RMS = 1.1298m.
- Horizontal RMS = 2.2522m. 3D RMS = 2.5196m.
- Mean Horizontal Distance = 1.9452m. Mean 3D Distance = 2.2691m.

These statistics are identical with those for Bursa-Wolf. This supports the argument in Section 2.9 that the Bursa-Wolf and Molodensky-Badekas are equivalent.

The 8-parameter affine transformation obtained by the new EEH-based method (described in Section 5.3) was of type “SR” with a Version-1 rotation matrix applied before scaling. The midpoints used to derive the local level coordinates were (3720212.60818, -157444.67338, 5147839.80851) at 54.3031795013°N, 2.4233924851°W in OSGB36, and (3720589.02187, -157555.97383, 5148271.46171) at 54.3033308221°N, 2.4248585239°W in WGS84. The parameters were as follows: $\Delta X'$ = 0m, $\Delta Y'$ = 0m, $\Delta Z'$ = 0m, $R_{X'}$ = 0.716361″, $R_{Y'}$ = 4.613041″, $R_{Z'}$ = 5.157975″, ΔS_h = -20.664894ppm and ΔS_v = -52.475000ppm. The main statistics of the residuals are:

- Latitude RMS = 1.6065m. Longitude RMS = 1.5898m. Height RMS = 1.0795m.
- Horizontal RMS = 2.2602m. 3D RMS = 2.5048m.
- Mean Horizontal Distance = 1.9540m. Mean 3D Distance = 2.2523m.

The 9-parameter affine transformation obtained by the new four-stage method described in Section 5.4 consisted of the following: ΔX = 574.154m, ΔY = -162.006m, ΔZ = 366.507m, R_X = -0.796459″, R_Y = -3.072092″, R_Z = 1.571086″, ΔS_X = -32.8663ppm, ΔS_Y = -15.8335ppm and ΔS_Z = 1.7736ppm. The main statistics of the residuals are:

- Latitude RMS = 1.5226m. Longitude RMS = 1.4361m. Height RMS = 1.1663m.
- Horizontal RMS = 2.0930m. 3D RMS = 2.3960m.
- Mean Horizontal Distance = 1.8353m. Mean 3D Distance = 2.1880m.

The 12-parameter affine transformation obtained using the method described in Section 4.12 consisted of 3 translation parameters and a multiplying matrix. The translation parameters were ΔX = 633.815m, ΔY = -425.804m and ΔZ = 645.324m. The matrix elements were

1 - 0.0000381588	-0.0000113448	-0.0000227724
0.0000307620	1 - 0.0000166821	0.0000383531
-0.0000103528	0.0000002387	1 - 0.0000340179

The main statistics of the residuals are:

- Latitude RMS = 1.3827m. Longitude RMS = 1.3600m. Height RMS = 1.0801m.
- Horizontal RMS = 1.9324m. 3D RMS = 2.2199m.
- Mean Horizontal Distance = 1.7298m. Mean 3D Distance = 2.0682m.

The new Standard Molodensky PCV transformation with 7 parameters obtained by the procedure described in Section 4.15 consisted of the following: ΔX_{hor} = 452.520, ΔY_{hor} = -134.223m, ΔZ_{hor} = 538.793m, R_Z = 1.091748″, ΔX_{ver} = 369.571m, ΔY_{ver} = -156.683m and ΔZ_{ver} = 434.664m. The main statistics of the residuals are:

- Latitude RMS = 1.6032m. Longitude RMS = 1.6039m. Height RMS = 1.0872m.
- Horizontal RMS = 2.2678m. 3D RMS = 2.5149m.
- Mean Horizontal Distance = 1.9561m. Mean 3D Distance = 2.2605m.

Compared with Standard Molodensky, the 3D RMS distance residual is reduced by 69.21% and the horizontal RMS residual is reduced by 71.69%. This is despite the fact that the application of the 7-parameter variation involves no extra computation apart from the addition of R_Z to the longitude shift.

The new Standard Molodensky PCV transformation with 6 parameters obtained by the procedure described in Section 4.15 consisted of the following: ΔX_{hor} = 453.370m, ΔY_{hor}

= -114.524m, ΔZ_{hor} = 538.810m, ΔX_{ver} = 369.571m, ΔY_{ver} = -156.683m and ΔZ_{ver} = 434.664m. The main statistics of the residuals are:

- Latitude RMS = 1.9818m. Longitude RMS = 1.9220m. Height RMS = 1.0872m.
- Horizontal RMS = 2.7608m. 3D RMS = 2.9671m.
- Mean Horizontal Distance = 2.4523m. Mean 3D Distance = 2.7218m.

Compared with Standard Molodensky, the 3D RMS distance residual is reduced by 63.68% and the horizontal RMS residual is reduced by 65.54%. This is despite the fact that the application of the 6-parameter variation involves no extra computation.

The new Abridged Molodensky PCV transformation with 7 parameters obtained by the procedure described in Section 4.16 consisted of the following: ΔX_{hor} = 452.265m, ΔY_{hor} = -134.191m, ΔZ_{hor} = 538.566m, R_Z = -1.090686″, ΔX_{ver} = 369.471m, ΔY_{ver} = -156.678m and ΔZ_{ver} = 434.664m. The main statistics of the residuals are:

- Latitude RMS = 1.6029m. Longitude RMS = 1.6006m. Height RMS = 1.0872m.
- Horizontal RMS = 2.2652m. 3D RMS = 2.5126m.
- Mean Horizontal Distance = 1.9543m. Mean 3D Distance = 2.2589m.

Compared with Abridged Molodensky, the 3D RMS distance residual is reduced by 69.18% and the horizontal RMS residual is reduced by 71.67%. This is despite the fact that the application of the 7-parameter variation involves no extra computation apart from the addition of R_Z to the longitude shift.

The new Abridged Molodensky PCV transformation with 6 parameters obtained by the procedure described in Section 4.16 consisted of the following: ΔX_{hor} = 453.114m, ΔY_{hor} = -114.511m, ΔZ_{hor} = 538.582m, ΔX_{ver} = 369.471m, ΔY_{ver} = -156.678m and ΔZ_{ver} = 434.664m. The main statistics of the residuals are:

- Latitude RMS = 1.9811m. Longitude RMS = 1.9185m. Height RMS = 1.0872m.
- Horizontal RMS = 2.7578m. 3D RMS = 2.9644m.
- Mean Horizontal Distance = 2.4501m. Mean 3D Distance = 2.7197m.

Compared with Abridged Molodensky, the 3D RMS distance residual is reduced by 63.64% and the horizontal RMS residual is reduced by 65.51%. This is despite the fact that the application of the 6-parameter variation involves no extra computation.

From these results, a few conclusions are offered:

- All the 3-parameter transformations derived from the 44-point dataset are very similar to the one derived from the 38-point dataset by DMA (NIMA, 2000, page B.5-4). This suggests that the former dataset is no worse and might be slightly more up-to-date.
- The 6-parameter variations on Standard and Abridged Molodensky reduce the 3D RMS residual by 64%, a significant benefit from separating ΔX_{ver}, ΔY_{ver} and ΔZ_{ver} from ΔX_{hor}, ΔY_{hor} and ΔZ_{hor}. Including R_Z as a 7th parameter in the Molodensky variations boosts that reduction to 69%.
- 7PC and its near-conformal versions improve on 3PC by virtually the same amount as the 7-parameter Standard and Abridged Molodensky variations. In this case, the 69% improvement is due to significant values for R_X, R_Y, R_Z and ΔS.
- Of the affine transformations, the 9-parameter and 12-parameter version give a significant improvement on 7PC, even taking into account the natural effect of extra parameters on RMSs of the residuals. This is not so of the 8-parameter affine transformation, despite the noticeably different values of ΔS_h and ΔS_v.

CHAPTER 10: CASE STUDY OF GHANA (3D)

This chapter covers the derivation of datum transformations from the Accra datum (based on the War Office ellipsoid) to WGS84. The area of application is Ghana's Golden Triangle, and there are 19 data points known in both datums. The coordinates are given in sub-appendix C.3.

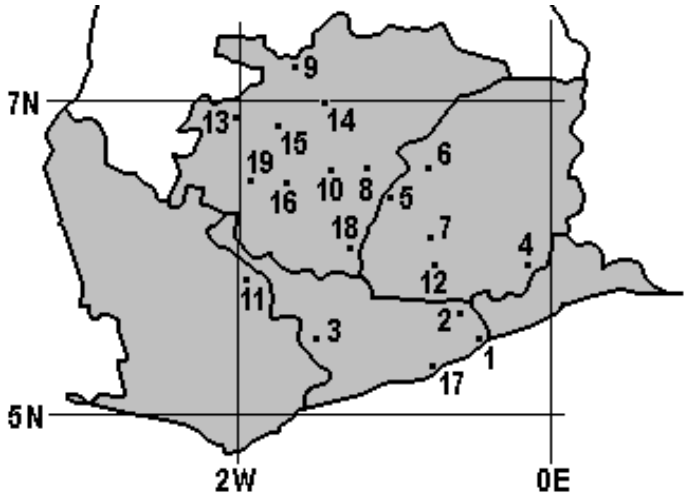


Figure 10-1: Data points for Ghana's Golden Triangle.

The location of the data points is shown in Figure 10-1. They are numbered 1 to 19 here, but are cross-referenced to official point identifications in sub-appendix C.3. It is clear from Figure 10-1 that the data points are too close to the line joining points 9 and 17 to give good coverage of the area as a whole.

The WGS84 coordinates are the result of a GPS survey (Dzidifo [2011], pages 43-44). From this, it appears that the WGS84 3D coordinates – and hence the geodetic heights – were obtained directly. The source of the latitudes and longitudes in the local system is "the Ghana Survey records, 1936", which also contain the orthometric heights (ibid, page 43). The estimation of the local *ellipsoidal* heights requires some explanation.

Ayer and Tiennah (2008) proposed a method for generating Accra ellipsoidal heights at 12 control points, where $(\phi_{Accra}, \lambda_{Accra})$ and $(\phi_{WGS84}, \lambda_{WGS84}, h_{WGS84})$ were known.

- The method of Molnár and Timár (2005), noted above in Section 4.5, was applied to derive values of ΔX, ΔY and ΔZ that gave the best least-squares fit to $(\phi_{Accra}, \lambda_{Accra}) \rightarrow (\phi_{WGS84}, \lambda_{WGS84})$ at the control points via Abridged Molodensky (*ie*, just using the formulae for $\Delta\phi$ and $\Delta\lambda$).

- Using the latest computed values of ΔX, ΔY and ΔZ, the Abridged Molodensky equation for Δh was applied to the Accra geodetic coordinates. The height shifts were applied to provisional values of h_{Accra} The differences between the computed and the known values of h_{WGS84} provided corrections to the provisional values of h_{Accra}.
- ΔX, ΔY and ΔZ were then recomputed using all the Abridged Molodensky equations (presumably from normal equations, but this is not made clear). The iterative approach "leads to progressive refinements in the values of Δh and the corresponding ΔX, ΔY and ΔZ until convergence is obtained".

The same approach was applied by Dzidifo (2011) to the 19 control points considered in this Chapter. The provisional values of h_{Accra} were set to zero. The values of ΔX, ΔY and ΔZ were found to be -196.7481, 32.7059 and 322.6385 respectively (ibid, Table 4.1). The derived values of h_{Accra} are given in sub-appendix C.3.

The view taken in this thesis is that the Accra ellipsoidal heights derived in this way are artificial and should not be used to deduce Accra geoid heights from the orthometric heights in Ghana's survey records. Ayer and Fosu (2008) did precisely that with the 12-point dataset, in a somewhat roundabout fashion. However, this study has not encountered any attempt to do this for the 19-point dataset.

The best argument for the estimated Accra ellipsoidal heights on the Accra Datum is that they facilitate the use of three-dimensional formulae to model horizontal datum shifts. Both Laari *et al* (2016) and Ziggah *et al* (2016) discuss accuracy of their derived 3D transformations purely in terms of horizontal accuracy. Nevertheless, a comprehensive range of 3D models has been derived for this thesis in pursuit of the following objectives:

- to see what levels of horizontal accuracy can be obtained from 3D methods on the Ghanaian data;
- to verify that the new methods of optimising non-linear transformations work on the Ghanaian dataset;
- to quantify the advantages of varying the Standard and Abridged Molodensky formulae as per 2.15 and 2.16.

As the dataset is one of the smaller ones acquired for this study, this case study is limited to transformations with a small number of parameters, thus ensuring plenty of degrees of freedom. All the data points were used as control points.

Individual transformations were derived by Excel VBA subroutines specifically written for this study, each of them based on least-squares optimisation in one form or another.

The 3-parameter conformal transformation obtained using the formulae in Section 4.3 consisted of the following: ΔX = -196.622m, ΔY = 33.361m and ΔZ = 322.344m. The main statistics of the residuals are:

- Latitude RMS = 0.9506m. Longitude RMS = 0.6600m. Height RMS = 0.0085m.
- Horizontal RMS = 1.1573m. 3D RMS = 1.1573m.
- Mean Horizontal Distance = 1.0578m. Mean 3D Distance = 1.0578m.

The Standard Molodensky transformation obtained using the formulae in Section 4.4 consisted of the following: ΔX = -196.614m, ΔY = 33.362m and ΔZ = 322.337m. The main statistics of the residuals are:

- Latitude RMS = 0.9506m. Longitude RMS = 0.6599m. Height RMS = 0.0082m.
- Horizontal RMS = 1.1572m. 3D RMS = 1.1572m.
- Mean Horizontal Distance = 1.0577m. Mean 3D Distance = 1.0577m.

If the parameters from optimising the 3PC model are used in Standard Molodensky (which is probably common practice), the 3D RMS distance residual is 1.1573m. The other values are within 0.0001m of those above, except for the height RMS which is 0.0011m.

NGA (2014, page D.2-1) gives the "Mean Solution" parameters for Accra to WGS84 as (-170±3)m, (33±4)m and (326±3)m. The number of satellite stations used was only 4, so the following qualification applies: "the 1σ errors for shift constants are non-computed estimates" (ibid, page D-5).

The Abridged Molodensky transformation obtained using the formulae in Section 4.5 consisted of the following: ΔX = -196.618m, ΔY = 33.360m and ΔZ = 322.433m. The main statistics of the residuals are:

- Latitude RMS = 0.9465m. Longitude RMS = 0.6598m. Height RMS = 0.0083m.
- Horizontal RMS = 1.1538m. 3D RMS = 1.1538m.
- Mean Horizontal Distance = 1.0550m. Mean 3D Distance = 1.0551m.

If the parameters from optimising the 3PC transformation are used in Abridged Molodensky (which is probably common practice), the 3D RMS distance residual is 8.1572m, which is 0.0035m worse. There are differences of up to 0.0041m in the other values.

The parameters differ slightly from those derived by Ziggah *et al* (2013, Table 1): -196.63455m, 33.36035m, 322.51867m.

The Helmert Version 1 transformation obtained by the new four-stage method (HO1 to HO4 in Section 5.1) consisted of the following: ΔX = -151.190m, ΔY = 31.593m, ΔZ = 327.177m, R_X = -0.445179″, R_Y = 0.005818″, R_Z = -0.021995″, ΔS = -7.167726ppm. The main statistics of the residuals are:

- Latitude RMS = 0.8421m. Longitude RMS = 0.4649m. Height RMS = 0.0076m.
- Horizontal RMS = 0.9619m. 3D RMS = 0.9619m.
- Mean Horizontal Distance = 0.8823m. Mean 3D Distance = 0.8824m.

The equivalent Helmert Version 2 transformation consisted of the following: ΔX = -151.190m, ΔY = 31.593m, ΔZ = 327.177m, R_X = -0.445179″, R_Y = 0.005818″, R_Z = -0.021995″, ΔS = -7.167726ppm. (In this case, the rotations are the same as for Version 1 when rounded to 6 decimal places.) The residuals are the same as for Helmert Version 1, so the main statistics are the same as for Version 1:

- Latitude RMS = 0.8421m. Longitude RMS = 0.4649m. Height RMS = 0.0076m.
- Horizontal RMS = 0.9619m. 3D RMS = 0.9619m.
- Mean Horizontal Distance = 0.8823m. Mean 3D Distance = 0.8824m.

The Bursa-Wolf transformation (fully-linear version) obtained using the formulae in Section 4.6 consisted of the following: ΔX = -151.190m, ΔY = 31.593m, ΔZ = 327.177m, R_X = -0.445176″, R_Y = 0.005818″, R_Z = -0.021995″, ΔS = -7.167757ppm. The main statistics of the residuals match those of Helmert Versions 1 and 2:

- Latitude RMS = 0.8421m. Longitude RMS = 0.4649m. Height RMS = 0.0076m.
- Horizontal RMS = 0.9619m. 3D RMS = 0.9619m.
- Mean Horizontal Distance = 0.8823m. Mean 3D Distance = 0.8824m.

The parameters agree almost exactly with those derived by Ziggah et al (2013, Table 1): -151.18907m, 31.59312m, 327.17669m, 0.44514″, -0.00582″, 0.02199″ and −7.16775ppm. The rotations are opposite in sign because that paper adopts the CF convention.

The Molodensky-Badekas transformation (fully-linear version) obtained using the formulae in Section 4.9 consisted of the following: ΔX = -196.622m, ΔY = 33.361m, ΔZ = 322.344m, R_X = -0.445176″, R_Y = 0.005818″, R_Z = -0.021995″, ΔS = -7.167757ppm; of no less importance are the values X_m = 6339126.3957m, Y_m = -133380.2931m, Z_m = 689482.7338m, obtained by averaging the Accra Cartesian coordinates. The main statistics of the residuals are:

- Latitude RMS = 0.8421m. Longitude RMS = 0.4649m. Height RMS = 0.0076m.
- Horizontal RMS = 0.9619m. 3D RMS = 0.9619m.
- Mean Horizontal Distance = 0.8823m. Mean 3D Distance = 0.8824m.

The parameters agree almost exactly with those derived by Ziggah *et al* (2013, Table 1): -196.62110m, 33.36129m, 322.34378m, 0.44514″, -0.00582″, 0.02199″ and −7.16775ppm. The rotations are opposite in sign because that paper adopts the CF convention.

The statistics of the residuals are identical with those for Bursa-Wolf. This supports the argument in Section 2.9 that the Bursa-Wolf and Molodensky-Badekas are equivalent.

The 8-parameter affine transformation obtained by the new EEH-based method (described in Section 5.3) was of type "SR" with a Version-1 rotation matrix applied before scaling. The midpoints used to derive the local level coordinates were (6339126.39671, -133380.29309, 689482.73388) at 6.2478954213°N, 1.2053712015°W in OSGB36, and (6338929.77460, -133346.93178, 689805.07751) at 6.2506680582°N, 1.2051071690°W in WGS84. The parameters were as follows: $\Delta X'$ = 0m, $\Delta Y'$ = 0m, $\Delta Z'$ = 0m, $R_{X'}$ = 9.975852″, $R_{Y'}$ = -0.918211″, $R_{Z'}$ = -0.548440″, ΔS_h = -7.167497ppm and ΔS_v = -4.440000ppm. The main statistics of the residuals are:

- Latitude RMS = 0.8420m. Longitude RMS = 0.4649m. Height RMS = 0.0080m.
- Horizontal RMS = 0.9618m. 3D RMS = 0.9619m.
- Mean Horizontal Distance = 0.8823m. Mean 3D Distance = 0.8823m.

The 9-parameter affine transformation obtained by the new four-stage method described in Section 5.4 consisted of the following: ΔX = 524.130m, ΔY = 45.719m, ΔZ = 256.376m, R_X = -0.505751″, R_Y=-2.279819″, R_Z = -0.485064″, ΔS_X = -112.447437ppm, $\Delta S_Y = -6.443500$ppm and ΔS_Z = -6.417050ppm. The main statistics of the residuals are:

- Latitude RMS = 0.8350m. Longitude RMS = 0.4713m. Height RMS = 0.0437m.
- Horizontal RMS = 0.9588m. 3D RMS = 0.9598m.
- Mean Horizontal Distance = 0.8831m. Mean 3D Distance = 0.8842m.

The 12-parameter affine transformation obtained using the method described in Section 4.12 consisted of 3 translation parameters and a multiplying matrix. The translation parameters were ΔX = 740.240m, ΔY = 1726.947m and ΔZ = -7387.552m. The matrix elements were

1 - 0.0001461491	-0.0000032467	-0.0000144420
-0.0002641697	1 - 0.0000023744	-0.0000279883
0.0012027468	-0.0000303308	1 + 0.0001181844

The main statistics of the residuals are:

- Latitude RMS = 0.6349m. Longitude RMS = 0.4352m. Height RMS = 0.0069m.
- Horizontal RMS = 0.7698m. 3D RMS = 0.7698m.
- Mean Horizontal Distance = 0.6599m. Mean 3D Distance = 0.6599m.

The matrix parameters derived in Ziggah *et al* (2013, Table 3) are similar but are given to a lower precision; the translation parameters are 739.98285m, 1726.7374m and 1587.1411m.

The new Standard Molodensky PCV transformation with 7 parameters obtained by the procedure described in Section 4.15 consisted of the following: ΔX_{hor} = -149.288m, ΔY_{hor} = 129.355m, $\Delta Z_{hor} =$ 327.292m, R_Z = -3.154177″, ΔX_{ver} = -196.629m, ΔY_{ver} = 33.361m and ΔZ_{ver} = 322.403m. The main statistics of the residuals are:

- Latitude RMS = 0.8524m. Longitude RMS = 0.4622m. Height RMS = 0.0004m.
- Horizontal RMS = 0.9696m. 3D RMS = 0.9696m.
- Mean Horizontal Distance = 0.8891m. Mean 3D Distance = 0.8891m.

Compared with Standard Molodensky, the 3D RMS distance residual is reduced by 16.22% and the horizontal RMS residual is reduced by 15.95%. This is despite the fact that the application of the 7-parameter variation involves no extra computation apart from the addition of R_Z to the longitude shift.

The new Standard Molodensky PCV transformation with 6 parameters obtained by the procedure described in Section 4.15 consisted of the following: ΔX_{hor} = -151.354m, ΔY_{hor} = 32.410m, ΔZ_{hor} = 327.293m, ΔX_{ver} = -196.629m, ΔY_{ver} = 33.361m and ΔZ_{ver} = 322.403m. The main statistics of the residuals are:

- Latitude RMS = 0.8687m. Longitude RMS = 0.4531m. Height RMS = 0.0004m.
- Horizontal RMS = 0.9798m. 3D RMS = 0.9798m.
- Mean Horizontal Distance = 0.8934m. Mean 3D Distance = 0.8934m.

Compared with Standard Molodensky, the 3D RMS distance residual is reduced by 15.34% and the horizontal RMS residual is reduced by 15.54%. This is despite the fact that the application of the 6-parameter variation involves no extra computation.

The new Abridged Molodensky PCV transformation with 7 parameters obtained by the procedure described in Section 4.16 consisted of the following: ΔX_{hor} = -149.449m, ΔY_{hor} = 129.320m, ΔZ_{hor} = 327.376m, R_Z = -3.185494″, ΔX_{ver} = -196.626m, ΔY_{ver} = 33.360m and ΔZ_{ver} = 322.432m. The main statistics of the residuals are:

- Latitude RMS = 0.8490m. Longitude RMS = 0.4628m. Height RMS = 0.0004m.
- Horizontal RMS = 0.9670m. 3D RMS = 0.9670m.
- Mean Horizontal Distance = 0.8867m. Mean 3D Distance = 0.8867m.

Compared with Abridged Molodensky, the 3D RMS distance residual is reduced by 16.19% and the horizontal RMS residual is reduced by 15.96%. This is despite the fact that the application of the 7-parameter variation involves no extra computation apart from the addition of R_Z to the longitude shift.

The new Abridged Molodensky PCV transformation with 6 parameters obtained by the procedure described in Section 4.16 consisted of the following: ΔX_{hor} = -151.536m, ΔY_{hor} = 32.412m, ΔZ_{hor} = 327.369m, ΔX_{ver} = -196.626m, ΔY_{ver} = 33.360m and ΔZ_{ver} = 322.432m. The main statistics of the residuals are:

- Latitude RMS = 0.8658m. Longitude RMS = 0.4533m. Height RMS = 0.0004m.
- Horizontal RMS = 0.9772m. 3D RMS = 0.9772m.
- Mean Horizontal Distance = 0.8911m. Mean 3D Distance = 0.8911m.

Compared with Abridged Molodensky, the 3D RMS distance residual is reduced by 15.31% and the horizontal RMS residual is reduced by 15.54%. This is despite the fact that the application of the 6-parameter variation involves no extra computation.

One conspicuous feature of all the derived transformations is that the RMS of the height residuals is less than 0.009m except for 9-parameter affine, so that the difference between the horizontal RMS and the 3D RMS is negligible. This is perhaps not surprising, given that the Accra ellipsoidal heights were created to extent a horizontal near-conformal best fit into a 3D near-conformal best fit, and even the affine transformations have "conformal" as a sub-type.

For practical purposes – given the issues over the heights – the transformations should be compared for *horizontal* accuracy. A few conclusions are offered:

- The 6-parameter variations on Standard and Abridged Molodensky reduce the horizontal RMS residual by 16%, a significant benefit from separating ΔX_{ver}, ΔY_{ver} and ΔZ_{ver} from ΔX_{hor}, ΔY_{hor} and ΔZ_{hor}.
- There is no significant benefit from including R_Z as a 7th parameter in the Molodensky variations, and it is clearly no coincidence that R_Z is small for Helmert (7PC) and Bursa-Wolf.
- 7PC and its near-conformal versions improve on 3PC by slightly more than the Standard and Abridged Molodensky variations. On the evidence of the derived parameters, this is largely due to differences in scale and the direction of the X-axis.
- Of the affine transformations, only the 12-parameter version gives a significant improvement on 7PC. It has the bonus of being easy to apply.

CHAPTER 11: CASE STUDY OF SWEDEN (3D)

This chapter covers the derivation of datum transformations from SWEREF93 (The Swedish realisation of ETRS89) to a local reference coordinate system designated RT90/RH70 by Andrei (2006). The latter is "a mixture of the Swedish triangulation network RT90 and the 2nd Swedish precise levelling network RH70". There are 20 data points known in both datums. The coordinates are given in sub-appendix C.4.

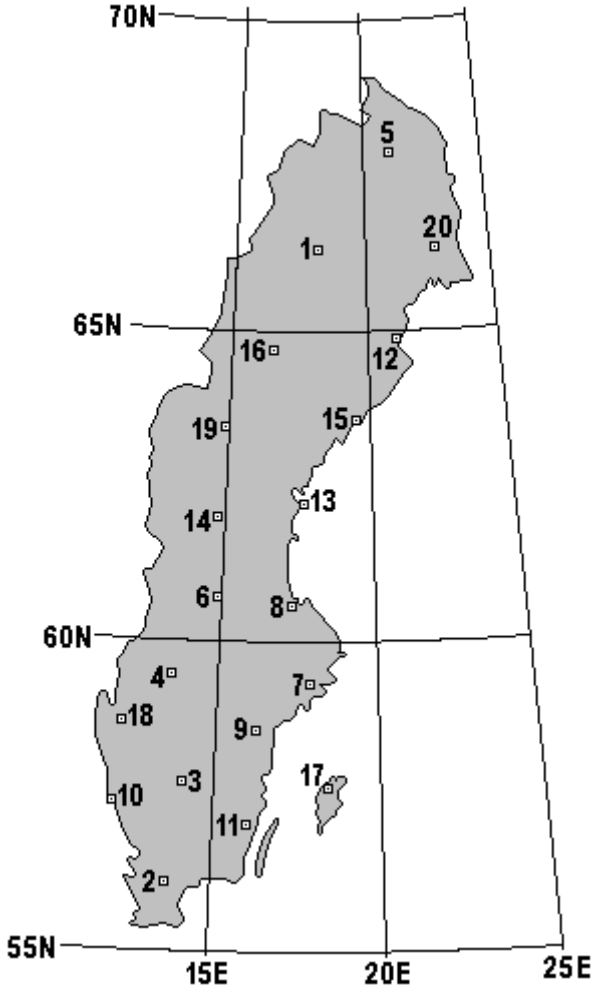


Figure 11-1: Data points for Sweden.

The location of the data points is shown in Figure 11-1. They are numbered 1 to 20 here, as in Andrei (2006) where no formal station names are given. From Jonsson *et al* (2003, Figure 1), it is clear that they are the 20 original stations of SWEPOS, the Swedish network of permanent reference stations, as at the start of 1994. The locations used as identifiers are listed in Figure C-4.

The age of the data means that neither set of ellipsoidal heights could have come directly from GNSS. As noted in sub-appendix C.4, it cannot be stated with certainty that the RT90/RH70

heights are ellipsoidal heights. When asked in 2019 on ResearchGate, Dr Andrei could not recall whether the heights had been converted rigorously to ellipsoidal heights via a local geoid model. Nevertheless, Andrei (2006, Section 1.3) does acknowledge the desirability of "taking into account the separation between the geoid and the ellipsoid of the local datum". Furthermore, Engberg and Lilje (2002) identifies RN92 as a geoid model to be used with RT90. The derivation of transformations from the RT90/RH70 Cartesian coordinates in both Andrei (2006) and Amiri-Simkooei (2018) is an implicit acceptance that the corresponding heights are ellipsoidal and that assumption is followed in this chapter.

As the dataset is one of the smaller ones acquired for this study, this case study is limited to transformations with a small number of parameters, thus ensuring plenty of degrees of freedom. All the data points were used as control points.

Individual transformations were derived by Excel VBA subroutines specifically written for this study, each of them based on least-squares optimisation in one form or another.

The 3-parameter conformal transformation obtained using the formulae in Section 4.3 consisted of the following: ΔX = -498.381m, ΔY = 36.616m and ΔZ = -563.444m. The main statistics of the residuals are:

- Latitude RMS = 5.1639m. Longitude RMS = 11.5094m. Height RMS = 5.8695m.
- Horizontal RMS = 12.6148m. 3D RMS = 13.9134m.
- Mean Horizontal Distance = 11.2544m. Mean 3D Distance = 12.3994m.

The Standard Molodensky transformation obtained using the formulae in Section 4.4 consisted of the following: ΔX = -498.396m, ΔY = 36.640m and ΔZ = -563.431m. The main statistics of the residuals are:

- Latitude RMS = 5.1664m. Longitude RMS = 11.5042m. Height RMS = 5.8693m.
- Horizontal RMS = 12.6111m. 3D RMS = 13.9100m.
- Mean Horizontal Distance = 11.2513m. Mean 3D Distance = 12.3965m.

If the parameters from optimising the 3PC transformation are used in Standard Molodensky (which is probably common practice), the 3D RMS distance residual is 13.9101m. The other values are within 0.0012m of those above, except for the means which are 11.2534m and 12.3985.

The Abridged Molodensky transformation obtained using the formulae in Section 4.5 consisted of the following: ΔX = -498.292m, ΔY = 36.665m and ΔZ = -563.445m. The main statistics of the residuals are:

- Latitude RMS = 5.1597m. Longitude RMS = 11.5038m. Height RMS = 5.8691m.
- Horizontal RMS = 12.6079m. 3D RMS = 13.9070m.
- Mean Horizontal Distance = 11.2495m. Mean 3D Distance = 12.3953m.

If the parameters from optimising the 3PC model are used in Abridged Molodensky (which is probably common practice), the 3D RMS distance residual is 13.9074m, which is 0.0004m higher. There are differences of up to 0.0063m in the other values.

The Helmert Version 1 transformation obtained by the new four-stage method (HO1 to HO4 in Section 5.1) consisted of the following: ΔX = -419.568m, ΔY = -99.246m, ΔZ = -591.456m, R_X = -0.850188″, R_Y = -1.814145″, R_Z = 7.853479″, ΔS = 1.023653ppm. The main statistics of the residuals are:

- Latitude RMS = 0.0615m. Longitude RMS = 0.1141m. Height RMS = 0.1243m.
- Horizontal RMS = 0.1296m. 3D RMS = 0.1796m.
- Mean Horizontal Distance = 0.1119m. Mean 3D Distance = 0.1665m.

The parameters are in strong agreement with Andrei's optimal Helmert Version 1 transformation. His parameters (Andrei, 2006, Table 4.3) are -419.568m, -99.246m, -591.456m, $0.850189''$, $1.814145''$, $-7.853479''$ and 1.0237ppm. The signs of the rotations are due to the CF rotation convention (in contrast to PV which was adopted for this study).

The optimal Helmert Version 2 transformation was obtained by computing the Version-2 rotations equivalent to the Version-1 rotations (as described in sub-section 2.6.1). The parameters consisted of the following: ΔX = -419.568m, ΔY = -99.246m, ΔZ = -591.456m, R_X = -0.850189″, R_Y = -1.814177″, R_Z = 7.853472″, ΔS = 1.023653ppm. The residuals are the same as for Helmert Version 1, so the main statistics are the same:

- Latitude RMS = 0.0615m. Longitude RMS = 0.1141m. Height RMS = 0.1243m.
- Horizontal RMS = 0.1296m. 3D RMS = 0.1796m.
- Mean Horizontal Distance = 0.1119m. Mean 3D Distance = 0.1665m.

The Bursa-Wolf transformation (fully-linear version) obtained using the formulae in Section 4.6 consisted of the following: ΔX = -419.571m, ΔY = -99.248m, ΔZ = -591.452m, R_X = -0.850184″, R_Y = -1.814094″, R_Z = 7.853516″, ΔS = 1.023087ppm. The main statistics of the residuals match those of Helmert Versions 1 and 2:

- Latitude RMS = 0.0615m. Longitude RMS = 0.1141m. Height RMS = 0.1243m.
- Horizontal RMS = 0.1296m. 3D RMS = 0.1796m.
- Mean Horizontal Distance = 0.1120m. Mean 3D Distance = 0.1665m.

The Molodensky-Badekas transformation (fully-linear version) obtained using the formulae in Section 4.9 consisted of the following: ΔX = -498.381m, ΔY = 36.616m, ΔZ = -563.444m, R_X = -0.850184″, R_Y = -1.814094″, R_Z = 7.853516″, ΔS = 1.023087ppm; of no less importance are the values X_m = 2943406.8346m, Y_m = 865099.156m, Z_m = 5558066.8176m, obtained by averaging the SWEREF93 Cartesian coordinates. The main statistics of the residuals are:

- Latitude RMS = 0.0615m. Longitude RMS = 0.1141m. Height RMS = 0.1243m.
- Horizontal RMS = 0.1296m. 3D RMS = 0.1796m.
- Mean Horizontal Distance = 0.1119m. Mean 3D Distance = 0.1665m.

The statistics of the residuals are identical with those for Bursa-Wolf (apart from a difference of 0.000066m in the mean horizontal distance). This supports the argument in Section 2.9 that the Bursa-Wolf and Molodensky-Badekas are equivalent.

The 8-parameter affine transformation obtained by the new EEH-based method (described in Section 5.3) was of type “SR” with a Version-1 rotation matrix applied before scaling. The midpoints used to derive the local level coordinates were (294306.8346, 865099.1656, 5558066.8176) at 61.26533543°N, 16.37863378°E in SWEREF93, and (2942908.45315, 865135.78170, 5557503.37315) at 61.26608350°N, 16.38191499°E in RT90/RH70. The parameters were as follows: $\Delta X'$ = 0m, $\Delta Y'$ = 0m, $\Delta Z'$ = 0m, $R_{X'}$ = 1.183752″, $R_{Y'}$ = -0.726807″, $R_{Z'}$ = -4.109541″, ΔS_h = 1.028061ppm and ΔS_v = -4.388300ppm. The main statistics of the residuals are:

- Latitude RMS = 0.0617m. Longitude RMS = 0.1153m. Height RMS = 0.1078m.
- Horizontal RMS = 0.1308m. 3D RMS = 0.1695m.
- Mean Horizontal Distance = 0.1129m. Mean 3D Distance = 0.1572m.

The parameters are in strong agreement with Andrei's optimal 8-parameter affine transformation. His parameters (Andrei, 2006, Table 4.4) are 0m, 0m, 0m, 1.813746″, -0.726803″, -4.109537″, 1.0281ppm and -4.3883m. The signs of the rotations are due to the CF rotation convention (in contrast to PV which was adopted for this study).

The 9-parameter affine transformation obtained by the new four-stage method described in Section 5.4 consisted of the following: ΔX = -422.590m, ΔY = -99.901m, ΔZ = -585.345m, R_X = -0.868570″, R_Y=-1.724599″, R_Z = 7.861213″, ΔS_X = 1.241500ppm, ΔS_Y = 1.080900ppm and ΔS_Z = 0.168000ppm. The main statistics of the residuals are:

- Latitude RMS = 0.0652m. Longitude RMS = 0.1138m. Height RMS = 0.1213m.
- Horizontal RMS = 0.1311m. 3D RMS = 0.1786m.
- Mean Horizontal Distance = 0.1125m. Mean 3D Distance = 0.1664m.

The parameters are in close agreement with Andrei's optimal 9-parameter affine transformation. His parameters (Andrei, 2006, Table 4.3) are -422.604m, -99.903m, -585.318m, $0.868641''$, $1.724187''$, $-7.861238''$ and 0.1642ppm. The signs of the rotations are due to the CF rotation convention (in contrast to PV which was adopted for this study).

The 12-parameter affine transformation obtained using the method described in Section 4.12 consisted of 3 translation parameters and a multiplying matrix. The translation parameters were ΔX = -414.166m, ΔY = -33.774m and ΔZ = -564.508m. The matrix elements were

1 + 0.0000006396	-0.0000382398	-0.0000095388
0.0000334154	1 - 0.0000004082	-0.0000049678
0.0000068835	-0.0000048371	1 - 0.0000027010

The main statistics of the residuals are:

- Latitude RMS = 0.0601m. Longitude RMS = 0.0442m. Height RMS = 0.1067m.
- Horizontal RMS = 0.0745m. 3D RMS = 0.1301m.
- Mean Horizontal Distance = 0.0648m. Mean 3D Distance = 0.1185m.

The new Standard Molodensky PCV transformation with 7 parameters obtained by the procedure described in Section 4.15 consisted of the following: ΔX_{hor} = -471.993m, ΔY_{hor} = -66.133m, ΔZ_{hor} = -569.643m, R_Z = 7.134725″, ΔX_{ver} = -416.328m, ΔY_{ver} = -99.283m and ΔZ_{ver} = -585.556m. The main statistics of the residuals are:

- Latitude RMS = 0.1115m. Longitude RMS = 0.1057m. Height RMS = 0.1293m.

- Horizontal RMS = 0.1536m. 3D RMS = 0.2008m.
- Mean Horizontal Distance = 0.1394m. Mean 3D Distance = 0.1852m.

Compared with Standard Molodensky, the 3D RMS distance residual is reduced by 98.56% and the horizontal RMS residual is reduced by 98.78%. This is despite the fact that the application of the 7-parameter variation involves no extra computation apart from the addition of R_Z to the longitude shift.

The new Standard Molodensky PCV transformation with 6 parameters obtained by the procedure described in Section 4.15 consisted of the following: ΔX_{hor} = -502.211m, ΔY_{hor} = 35.655m, ΔZ_{hor} = -569.957m, ΔX_{ver} = -416.328m, ΔY_{ver} = -99.283m and ΔZ_{ver} = -585.556m. The main statistics of the residuals are:

- Latitude RMS = 4.8139m. Longitude RMS = 11.6407m. Height RMS = 0.1293m.
- Horizontal RMS = 12.5968m. 3D RMS = 12.5975m.
- Mean Horizontal Distance = 11.2436m. Mean 3D Distance = 11.2448m.

Compared with Standard Molodensky, the 3D RMS distance residual is reduced by 9.44%, despite the fact that the application of the 6-parameter variation involves no extra computation. The improvement is due to better modelling of the height shift, since the horizontal RMS residual is reduced by only 0.11%.

The new Abridged Molodensky PCV transformation with 7 parameters obtained by the procedure described in Section 4.16 consisted of the following: ΔX_{hor} = -471.841m, ΔY_{hor} = -66.052m, ΔZ_{hor} = -569.542m, R_Z = 7.132244″, ΔX_{ver} = -416.227m, ΔY_{ver} = -99.255m and ΔZ_{ver} = -585.569m. The main statistics of the residuals are:

- Latitude RMS = 0.1127m. Longitude RMS = 0.1050m. Height RMS = 0.1292m.
- Horizontal RMS = 0.1540m. 3D RMS = 0.2010m.
- Mean Horizontal Distance = 0.1399m. Mean 3D Distance = 0.1860m.

Compared with Abridged Molodensky, the 3D RMS distance residual is reduced by 98.55% and the horizontal RMS residual is reduced by 98.78%. This is despite the fact that the application of the 7-parameter variation involves no extra computation apart from the addition of R_Z to the longitude shift.

The new Abridged Molodensky PCV transformation with 6 parameters obtained by the procedure described in Section 4.16 consisted of the following: ΔX_{hor} = -502.047m, ΔY_{hor} = 35.698m, ΔZ_{hor} = -569.856, ΔX_{ver} = -416.227m, ΔY_{ver} = -99.255m and ΔZ_{ver} = -585.569m. The main statistics of the residuals are:

- Latitude RMS = 4.8131m. Longitude RMS = 11.6379m. Height RMS = 0.1292m.
- Horizontal RMS = 12.5940m. 3D RMS = 12.5946m.
- Mean Horizontal Distance = 11.2421m. Mean 3D Distance = 11.2432m.

Compared with Abridged Molodensky, the 3D RMS distance residual is reduced by 9.44%, despite the fact that the application of the 6-parameter variation involves no extra computation. The improvement is due to better modelling of the height shift, since the horizontal RMS residual is reduced by only 0.11%.

From these results, a few conclusions are offered:

- The 3-parameter transformations are unsuited to this dataset, giving an RMS value of 13.9m for the residuals. It is surely significant that all transformations between RT90 and geocentric datums encountered during this study have at least 6 parameters including 3 rotations (Andrei [2006], Amiri-Simkooei [2018] and ESRI [2012]).
- The 6-parameter variations on Standard and Abridged Molodensky reduce the 3D RMS residual by 9.4%, almost entirely due to a better fit to the height data. Including R_Z as a 7th parameter in the Molodensky variations improves that reduction to 98.6% because there are more than 7 arc-seconds between the zero meridians.
- 7PC and its near-conformal versions improve on 3PC to the extent of reducing the 3D RMS residual by 98.7%. In this case, the improvement is due to significant values for R_X, R_Y, R_Z (especially) and ΔS.
- Of the affine transformations, the 9-parameter version gives no real improvement on 7PC, but the 8-parameter version is better than both. This vindicates Andrei's use of local level coordinates to separate ΔS_h and ΔS_v. It does so rather more than his own paper which claimed no improvements in the residuals.
- The 12-parameter affine transformation gives the best fit of all (3D RMS 0.13m). While extra parameters would be expected to have a downward effect on the residuals, in this case the extent of the reduction suggests that actual departures from conformality have been modelled.

CHAPTER 12: CASE STUDY OF SLOVENIA (2D)

This chapter covers the derivation of datum transformations from local datum D48 to D96, the latter being a locally-adopted name for ETRS89 (which is a global datum). The area of application is Slovenia and there are 3331 data points known in both datums. Their distribution was analysed computationally for this study and is illustrated in Figure 12-1; numbers indicate the number of points in each 20km square. More details of the datasets are given in sub-appendix C.5.

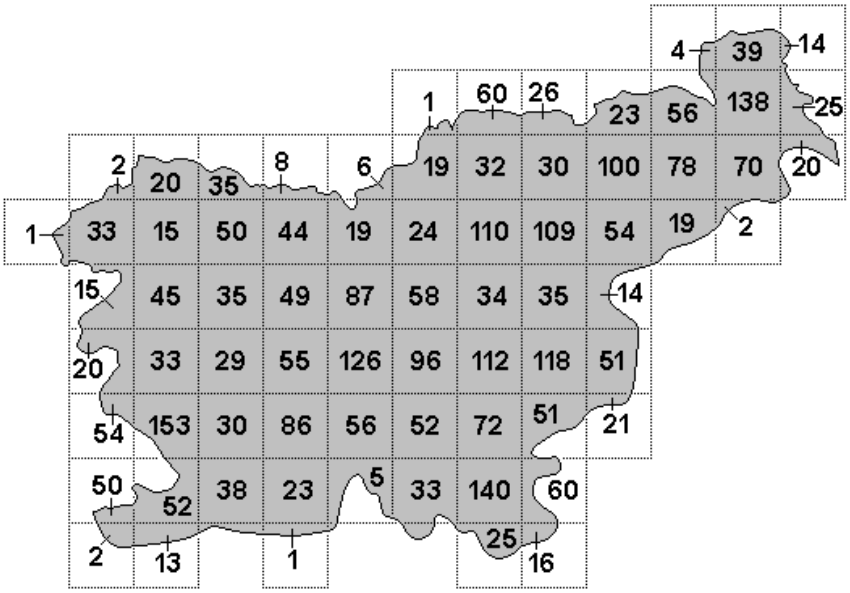


Figure 12-1: Distribution of data points for Slovenia, in terms of numbers per 20km-square.

The points are vertices of a triangulation designed for rubber sheeting. Similarity transformations were deemed unsuitable due to "varying scale" in "the old system" (Berk and Komadina [2013]). By contrast, the vast majority of the D96 coordinates were based on GNSS surveys (directly), according to Sandi Berk when asked in 2021 on ResearchGate; a few eccentric points were measured to achieve better measurement conditions. It was noted in Section 7.3 that there were also virtual tie points whose coordinates are not observations in the conventional sense.

As converted for use in this study, the coordinates are (ϕ, λ) in the respective datums. Given the two-dimensional nature of the dataset and the known scale distortion, the types of transformation considered were multiple regression equations and composite methods.

In both cases the 3331 points in the dataset were split into:

- 3123 control points from which the transformations were derived, and
- 208 test points to be used for an independent check on accuracy.

For this study, the algorithm to select test points from the data points was as follows. Grid points were created, roughly 9km apart (more precisely, 288″ of latitude, 432″ of longitude). A 158.4″-by-236.4″ rectangle was put round each grid point, ensuring at least 4km between any two rectangles (more precisely, 129.6″ of latitude 194.4″ of longitude). The first data point within each rectangle was selected as a test point. This is illustrated in Figure 12-2.

The selection process was designed to ensure that the test points are well spread out. The "corridors" ensure that all selected points are at least 4km apart. This method has the potential drawback of removing an isolated data point from the control points. The distribution of data points shown in Figure 12-1 suggests that the risk of that is very small.

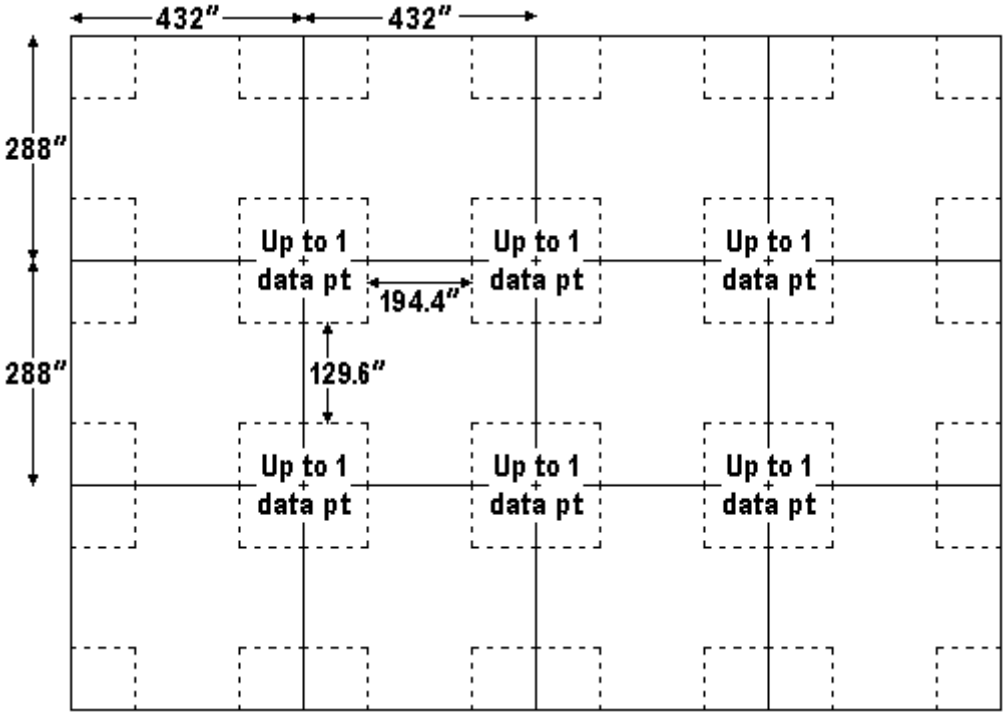


Figure 12-2: Selection of test points from the data points in the dotted rectangles.

12.1 Application of multiple regression equations

The multiple regression equations considered were the fully-normalised MREs described in Section 2.17. That is to say that the intermediate coordinates U and V were defined in such a way that they varied between -1 and 1 over Slovenia.

Only selected MREs are included in this Section. The full list of MREs can be found in sub-appendix H.2.

The offset coordinates in degrees were 46.150 for latitude and 14.984 for longitude. The relative latitude and relative longitude were scaled as follows:

$$U = 1.37174\left(\phi_{\text{in deg}} - 46.150\right); \tag{12-1}$$
$$V = 0.62035\left(\lambda_{\text{in deg}} - 14.984\right). \tag{12-2}$$

The relationship between the geodetic and intermediate coordinates is illustrated in Figure 12-3.

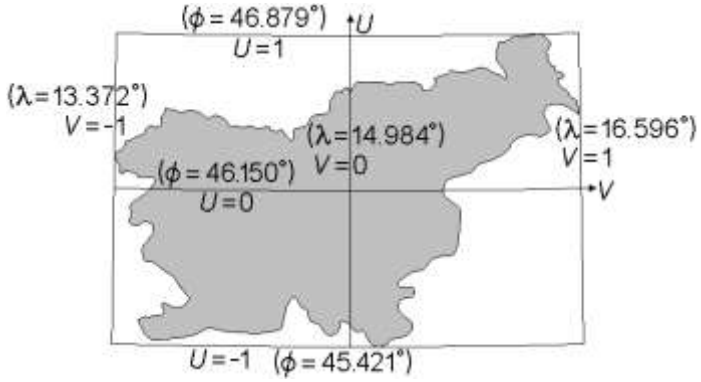


Figure 12-3: The relationship between (ϕ, λ) and (U, V).

The number of points in the dataset meant there was no need for the limit of 40 terms per MRE imposed on Western Australia. However, to keep the size of the expressions manageable, it was decided to set a limit of roughly 50 terms. Taking account of Tables 2-4 to 2-8, this meant setting upper limits on the top power as follows:

- 6 for Ordinary MREs and Chebyshev MREs;
- 4 for North/South MREs, East/West MREs and Four-Quadrant MREs.

The recognised approach for deriving MREs is to apply least-squares optimisation on different combination of terms and to retain those terms which are statistically significant. The implementation of that approach for this study is the one described in Section 5.5: "eliminating ratios less than one" (ERLTO), where the term "ratio" is the absolute value of a parameter divided by its standard error (AP/SE). The full list of actual MREs is given in sub-appendix H.2.

The Ordinary MREs with top power 3 obtained by ERLTO had 13 terms for the latitude shift and 14 for the longitude shift. The RMSs of the residuals at the test points are 0.1022m for

latitude, 0.0981m for longitude and 0.1417m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0887m, 0.0906m and 0.1268m.)

The Ordinary MREs with top power 4 obtained by ERLTO had 25 terms for the latitude shift and 22 for the longitude shift. The RMSs of the residuals are 0.0968m for latitude, 0.0896m for longitude and 0.1320m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0804m, 0.0826m and 0.1153m.)

The Ordinary MREs with top power 5 obtained by ERLTO had 34 terms for the latitude shift and (purely by coincidence) 34 for the longitude shift. The RMSs of the residuals are 0.0893m for latitude, 0.0817m for longitude and 0.1210m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0728m, 0.0732m and 0.1032m.)

The Ordinary MREs with top power 6 obtained by ERLTO had 46 terms for the latitude shift and 45 for the longitude shift. The RMSs of the residuals are 0.0851m for latitude, 0.0788m for longitude and 0.1160m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0688m, 0.0690m and 0.0974m.)

The North/South MREs with top power 3 obtained by ERLTO had 24 terms for the latitude shift and (purely by coincidence) 24 for the longitude shift. The RMSs of the residuals are 0.0975m for latitude, 0.0931m for longitude and 0.1348m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0807m, 0.0852m and 0.1174m.)

> Compared with Ordinary MREs with top power 4, the accuracy is 2.1% worse and requires 1 more term.

The North/South MREs with top power 4 obtained by ERLTO had 35 terms for the latitude shift and 39 for the longitude shift. The RMSs of the residuals are 0.0901m for latitude, 0.0877m for longitude and 0.1257m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0725m, 0.0773m and 0.1060m.)

Compared with Ordinary MREs with top power 5, the accuracy is 3.9% worse and requires 6 more terms.

The East/West MREs with top power 3 obtained by ERLTO had 22 terms for the latitude shift and (purely by coincidence) 22 for the longitude shift. The RMSs of the residuals are 0.0954m for latitude, 0.0882m for longitude and 0.1299m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0824m, 0.0812m and 0.1157m.)

Compared with Ordinary MREs with top power 4, the accuracy is 1.6% better and was achieved with 3 fewer terms.

The East/West MREs with top power 4 obtained by ERLTO had 34 terms for the latitude shift and 36 for the longitude shift. The RMSs of the residuals are 0.0897m for latitude, 0.0778m for longitude and 0.1187m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0718m, 0.0716m and 0.1014m.)

Compared with Ordinary MREs with top power 5, the accuracy is 1.9% better although it required 2 more terms. For the longitude shift, the improvement was 4.8% although again it required 2 more terms.

The Four-Quadrant MREs with top power 3 obtained by ERLTO had 27 terms for the latitude shift and 28 for the longitude shift. The RMSs of the residuals are 0.0928m for latitude, 0.0883m for longitude and 0.1281m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0752m, 0.0801m and 0.1099m.)

Compared with Ordinary MREs with top power 4, the accuracy is 3.0% better but was achieved with 8 more terms.

The Four-Quadrant MREs with top power 4 obtained by ERLTO had 48 terms for the latitude shift and 50 for the longitude shift. The RMSs of the residuals are 0.0888m for latitude, 0.0780m for longitude and 0.1182m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0709m, 0.0699m and 0.0995m.)

Compared with Ordinary MREs with top power 6, the accuracy is 1.9% worse and requires 9 more terms.

The Chebyshev MREs with top power 3 obtained by ERLTO had 13 terms for the latitude shift and 15 for the longitude shift. The RMSs of the residuals at the test points are 0.1022m for latitude, 0.0979m for longitude and 0.1414m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0887m, 0.0906m and 0.1268m.)

Compared with Ordinary MREs with top power 3, the accuracy is much the same (0.2% better) but requires 1 more term.

The Chebyshev MREs with top power 4 obtained by ERLTO had 24 terms for the latitude shift and (purely by coincidence) 24 for the longitude shift. The RMSs of the residuals at the test points are 0.0968m for latitude, 0.0898m for longitude and 0.1321m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0805m, 0.0826m and 0.1153m.)

Compared with Ordinary MREs with top power 4, the accuracy is much the same (0.1% better) but requires 1 more term.

The Chebyshev MREs with top power 5 obtained by ERLTO had 34 terms for the latitude shift and 32 for the longitude shift. The RMSs of the residuals at the test points are 0.0893m for latitude, 0.0814m for longitude and 0.1208m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0728m, 0.0732m and 0.1032m.)

Compared with Ordinary MREs with top power 5, the accuracy is much the same (0.3% better) but requires 2 fewer terms.

The Chebyshev MREs with top power 6 obtained by ERLTO had 49 terms for the latitude shift and 46 for the longitude shift. The RMSs of the residuals at the test points are 0.0852m for latitude, 0.0789m for longitude and 0.1161m for horizontal distance. (These are larger

than the corresponding RMS differences at the control points, which were 0.0688m, 0.0689m and 0.0974m.)

Compared with Ordinary MREs with top power 6, the accuracy is much the same (just 0,1% worse) but requires 4 more terms.

Of the ordinary MREs obtained, the most accurate was the pair with top power 6, denoted "Ord6" for convenience. Of the piecewise MREs obtained (N/S, E/W, 4Q), the most accurate was the E/W pair with top power 4, denoted "EW4" for convenience. For Ord6, the RMS of the test-point residuals was 0.0851m for latitude (from 46 terms) and 0.0788m for longitude (from 45 terms). For EW4, the RMS of the test-point residuals was 0.0897m for latitude (from 34 terms) and 0.0778m for longitude (from 36 terms).

To simplify comparisons between Ord6 and EW4, additional trimming (as described in Section 5.5) was applied. The least significant terms were discarded from Ord6(Lat) until the accuracy matched that of EW4(Lat). The least significant terms were discarded from EW4(Lon) until the accuracy matched that of Ord6(Lon). The trimmed Ord6(Lat), denoted "Ord6tr(Lat)", had 31 terms compared with 34 for EW4(Lat). The trimmed EW4(Lon), denoted "EW4tr(Lon)", had 27 terms compared with 45 for Ord6(Lon).

The trimmed pairing of Ord6tr(Lat) and EW4tr(Lon) has RMS test-point residuals 0.0895m for latitude, 0.0789m for longitude and 0.1193m for horizontal distance. (These are larger than the corresponding RMS differences at the control points, which were 0.0731m, 0.0737m and 0.1038m.) Compared with Ordinary MREs with top power 6, the accuracy is 2.8% worse but is achieved with 33 fewer terms; it is therefore a good compromise between accuracy and economy. The equations of the trimmed pairing are:

$$\begin{aligned}\Delta\phi('') = &-1.09184 - 0.25923U + 0.13427V + 0.01258V^2 - \\ &0.02234U^3 + 0.11929U^2V - 0.13431UV^2 + \\ &0.09536V^3 + 0.05600U^4 - 0.11519U^3V - \\ &0.28478UV^3 - 0.25322U^4V + 0.29737U^3V^2 - \\ &0.52256U^2V^3 + 0.44093UV^4 - 0.17377V^5 - \\ &0.05652U^6 + 0.37998U^3V^3 - 0.39517U^2V^4 + \\ &0.58467UV^5 - 0.11294V^6 + 1.33082U^4V^3 - \\ &1.35697U^3V^4 - 0.33803U^5V^3 + 0.75580U^4V^4 - \\ &0.50114U^6V^3 + 1.04331U^3V^6 + 1.01810U^6V^4 - \\ &1.65939U^5V^5 + 1.22406U^6V^5 - 1.63569U^5V^6\end{aligned} \qquad (12\text{-}3)$$

$$\Delta\lambda('') = \begin{cases} -17.24931 - 0.13101U - 0.01532U^2 + \\ 0.02545U^3 + 0.05804U^4 - 0.62942V - \\ 0.12471U^2V + 0.11838U^3V + 0.32602U^4V \\ -0.18680V^2 + 0.08986UV^2 + 1.41112U^2V^2 - \\ 0.55717U^3V^2 - 2.9331U^4V^2 + 0.07985V^3 - \\ 2.48610U^2V^3 + 5.56784U^4V^3 + \\ 1.23345U^2V^4 - 2.65893U^4V^4 \end{cases} \text{ if } V>0$$

$$\begin{cases} +0.30075V^2 - 0.03472UV^2 + 0.40479U^4V^2 + \\ 0.64266V^3 - 0.68551U^3V^3 + 0.38714V^4 - \\ 0.34671U^2V^4 - 0.74364U^3V^4 \end{cases} \text{ if } V\leq 0 \qquad (12\text{-}4)$$

The Chebyshev MREs in sub-appendix H.2 show a distinct tendency to have smaller coefficients than the MREs based on monomials. The significance of this can be deduced by comparing the initial ordinary MRE with top power 6 with the initial Chebyshev MRE with top 6; "initial" means "prior to application of ERLTO" so that they are different representations of the same polynomial. The L_1 norms of the coefficients were as follows:

- for the latitude shift, 54.8787 (Ord) and 14.7254 (Cheb);
- for the longitude shift, 111.4905 (Ord) and 37.5481 (Cheb).

Equivalence of the functions means that the lower Chebyshev L_1 norms are limits on the magnitudes of the ordinary MREs. (See subsections 2.17.1 and 2.17.5.) This implies that some of the seemingly-large monomial terms largely cancel each other out. It is logical to assume that this is true of the monomial-based MREs in general, including N/S, E/W and 4Q. This revelation is perhaps the most significant contribution of Chebyshev polynomials to the study of MREs in Slovenia.

Contour maps of most of the MREs are shown in in sub-appendix I.2. They show strong similarities to one another, with no sharp bends or major variations in spacing.

The main finding of this section is that East/West MREs work well for transforming D48 to D96. The ordinary MRE in (12-3) and the East/West MRE in (12-4) offer perhaps the best pairing for accuracy and economy.

12.2 Application of composite methods

The composite methods applied in Slovenia were selected from those described in Chapter 7 and the methods for interpolation of gridded data were selected from those described in Chapter 6. The generic process used is illustrated in Figure 12-4.

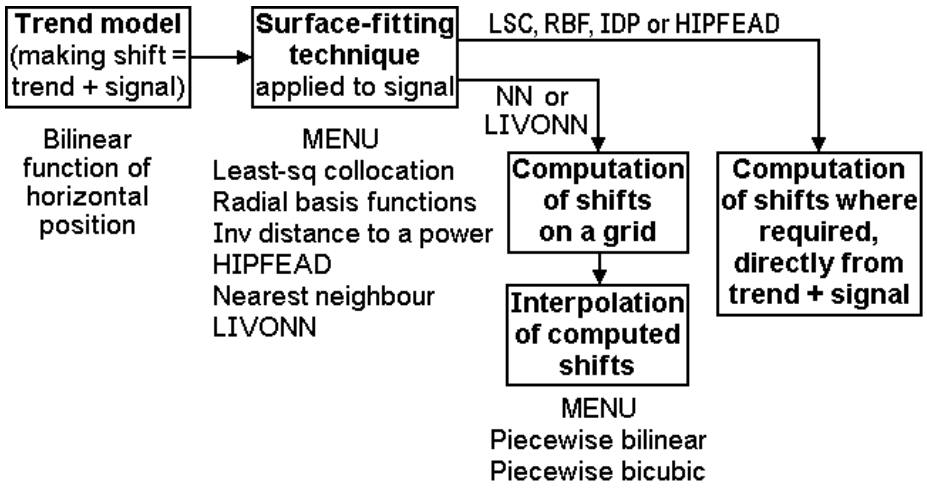


Figure 12-4: Composite methods chosen for D48→D96 in Slovenia.

The trend model chosen for the latitude shift (in arc-seconds) was

$$\Delta\phi_{\mathrm{BLM}}('') = -1.09103 - 0.37669\left(\phi_{\mathrm{in\ deg}} - 46.150\right) + 0.091185\left(\lambda_{\mathrm{in\ deg}} - 14.984\right) - 0.039774\left(\phi_{\mathrm{in\ deg}} - 46.150\right)\left(\lambda_{\mathrm{in\ deg}} - 14.984\right). \quad (12\text{-}5)$$

The trend model chosen for the longitude shift (in arc-seconds) was

$$\Delta\lambda_{\mathrm{BLM}}('') = -17.25104 - 0.18793\left(\phi_{\mathrm{in\ deg}} - 46.150\right) - 0.41814\left(\lambda_{\mathrm{in\ deg}} - 14.984\right) - 0.00284\left(\phi_{\mathrm{in\ deg}} - 46.150\right)\left(\lambda_{\mathrm{in\ deg}} - 14.984\right). \quad (12\text{-}6)$$

In each case, the trend model is the bilinear model that gives the least-squares fit to the shifts at the control points, hence the subscript BLM. This model is mathematically equivalent to the multiple regression equation with top power 1. The bilinear form has sufficient terms to ensure that

- the trend model includes the average shift as a constant term, and
- embodies the tilt which is evident from the contour maps I-34 to I-42 in Appendix I.

For both the latitude shift and the longitude shift, the signal component is that part which is unmodelled.

The surface-fitting techniques tested for each signal were those shown in Figure 12-4. They are also listed below with the corresponding Sections.

- Least-squares collocation (7.1);
- Radial basis functions (7.6);
- Inverse distance to a power (7.5);
- Hybrid inverse power function embodying accelerated decline [HIPFEAD] (7.13);
- Nearest neighbour (7.7);
- Linear interpolation variation on nearest neighbour [LIVONN] (7.14).

Figure 12-4 differs from Figure 8-3 in that the emphasis has shifted from SFTs as a gridding method to SFTs as an approximation method to be applied wherever required. The first four techniques generate a C^1 surface and do not require a huge amount of computation. It is true that least-squares collocation and radial basis functions require a one-off computation of revamped signals from a large matrix equation. However, this would be the case if the methods were employed as a gridding method. The difference is that direct application requires storage of the revamped signals at control points instead of the signal values at grid points. The direct approach has the advantage that it interpolates exact data values rather than approximate values at grid points.

The SFTs listed which do not generate a C^1 surface are nearest neighbour and LIVONN. Nearest neighbour creates discontinuities and LIVONN is liable to do so in places. These were treated as gridding methods to be followed by bilinear interpolation (Section 6.2) or bicubic interpolation (6.3, as in 6.3.3). The grid was defined by intervals of 72″ (0.02°).

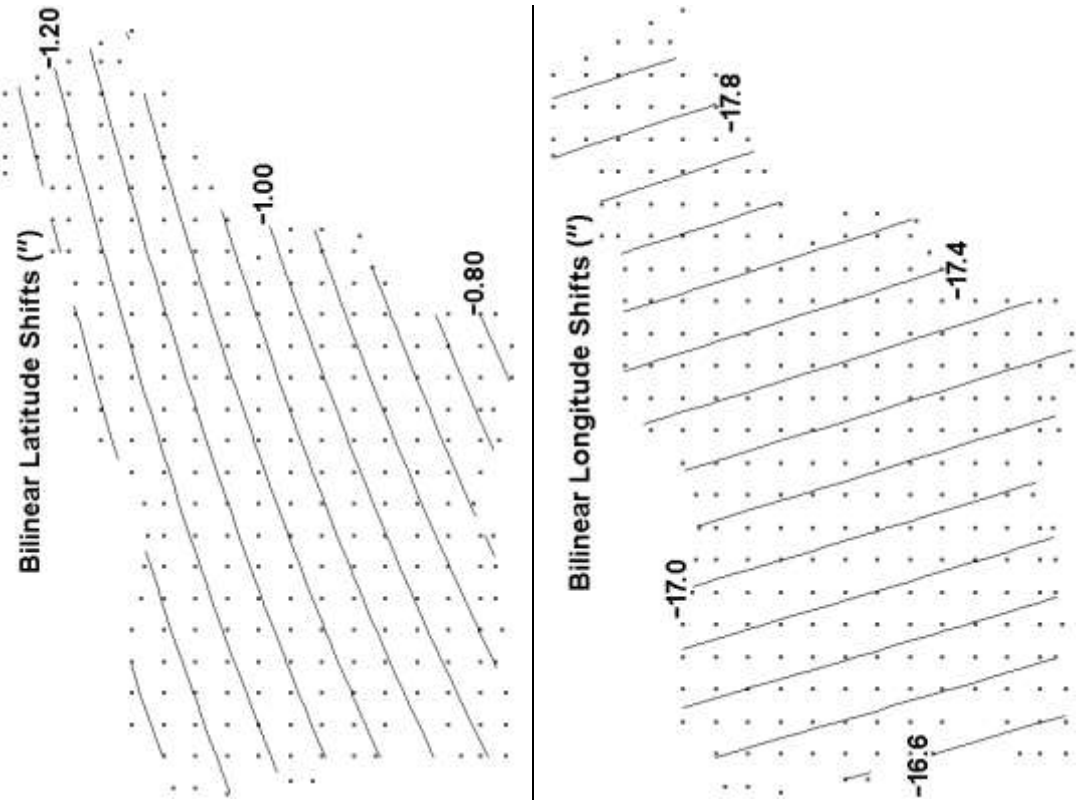


Figure 12-5: Contour maps of the trend models of D48→D96 in arc-seconds.

Figure 12-5 shows the contours generated by LSS for the trend model defined by (12-5) and (12-6). (The points used in these maps are considerably fewer than the 72″ grid of computation points.)

Least-squares collocation was applied with three covariance functions applied to the signals. These were Gaussian, SS20 and SS30, described in subsection 7.1. Variance (in square metres) was 0.0176595 for the latitude signal and 0.0406728 for the longitude signal. In both cases this was computed from the control points.

The size of the dataset made it possible to estimate the correlation length (and hence the 2nd constant required by each covariance function) by sampling at the control points. Tables 12-1 and 12-2 feature the ranges of distance for which the estimated correlation was close to 0.5.

Table 12-1: Sampling of latitude-signal products to estimate covariance and correlation

Min of Range	Max of Range	Centre of Range	Sample Size	Estimated Covariance From Mean	Estimated Correlation From Mean
13300	13500	13400	3687	0.0093389	0.528834
13500	13700	13600	3754	0.0089711	0.508006
13700	13900	13800	3820	0.0090454	0.512211
13900	14100	14000	3714	0.0086743	0.491200
14100	14300	14200	3784	0.0087971	0.498153
14300	14500	14400	3835	0.0089585	0.507289
14500	14700	14600	3699	0.0083238	0.471352
14700	14900	14800	3910	0.0088254	0.499752
14900	15100	15000	4000	0.0085003	0.481343

The linear function giving the least-squares fit to the data in columns 3 and 6 is $\text{Correlation} = 0.536551 - 0.00000258537 \times \text{Centre}$. Substitution of 0.5 for Correlation gives a value of 14120.06 for Centre, so the estimated correlation length for the latitude signal is 14120m.

Table 12-2: Sampling of longitude-signal products to estimate covariance and correlation

Min of Range	Max of Range	Centre of Range	Sample Size	Estimated Covariance From Mean	Estimated Correlation From Mean
11900	12100	12000	3473	0.0215594	0.530069
12100	12300	12200	3427	0.0198642	0.488391
12300	12500	12400	3469	0.0208016	0.511438
12500	12700	12600	3628	0.0203653	0.500711
12700	12900	12800	3587	0.0205790	0.505965
12900	13100	13000	3549	0.0214277	0.526832
13100	13300	13200	3673	0.0195208	0.479948
13300	13500	13400	3687	0.0196559	0.483270
13500	13700	13600	3754	0.0190541	0.468473

The linear function giving the least-squares fit to the data in columns 3 and 6 is $\text{Correlation} = 0.817968 - 0.0000248838 \times \text{Centre}$. Substitution of 0.5 for Correlation gives a value of 12778.11 for Centre, so the estimated correlation length for the longitude signal is 12778m.

The conversion of correlation length into the covariance function's 2nd constant (k or r_{max}) is described in Examples 7.1 to 7.3 in Chapter 7. The covariance functions to be adopted were therefore as follows:

- Gaussian, as per formula (7-11): for the latitude signal, $C_0 = 0.0176595$ and $k = 1/16970$; for the longitude signal, $C_0 = 0.0406728$ and $k = 1/15348$.
- SS20 as per formula (7-14): for the latitude signal, $C_0 = 0.0176595$ and $r_{max} = 39095$; for the longitude signal, $C_0 = 0.0406728$ and $r_{max} = 35379$.
- SS30 as per formula (7-17): for the latitude signal, $C_0 = 0.0176595$ and $r_{max} = 37682$; for the longitude signal, $C_0 = 0.0406728$ and $r_{max} = 34102$.

When the Gaussian covariance function was used, least-squares collocation failed. Initially, Cholesky decomposition as per sub-appendix G.2.2 was used to solve equation (7-29) which should have provided the revamped signal vector. The process broke down ("the term to be square-rooted isn't positive") in Row 88 for the latitude revamped signal vector and row 99 for the longitude revamped signal vector. The matrix-inversion method of sub-appendix G.2.3 did provide "solutions", but substitution into (7-29) gave misclosures of up to 582.781m in latitude and 57.694m in longitude. The covariance matrices relating the revamped signals to the control-point signals were obviously ill-conditioned.

When the SS20 covariance function was used, least-squares collocation failed, although less spectacularly. Initially, Cholesky decomposition as per sub-appendix G.2.2 was used to solve equation (7-29) which should have provided the revamped signal vector. The process broke down ("the term to be square-rooted isn't positive") in Row 122 for both the latitude revamped signal vector and the longitude revamped signal vector. The matrix-inversion method of sub-appendix G.2.3 did provide solutions that satisfied the solution-check process, although the values of the revamped signals were suspiciously large. However, application of (7-27) to compute signals at the test points produced model signals varied from 52.896m to -61.754m for latitude and 39.959m to -32.605m for longitude. Clearly the surfaces fitted to the control points were far too volatile.

When the SS30 covariance function was used, least-squares collocation was successful. Cholesky decomposition as per sub-appendix G.2.2 was used to solve equation (7-29). Both the latitude revamped signal vector and the longitude revamped signal vector satisfied the solution-check process. Application of (7-27) at the test points produced model signals close to the actual signals. RMS differences were 0.1172m for latitude, 0.0795m for longitude and 0.1416m for horizontal distance.

Radial basis functions were applied using all the examples described in subsection 7.6: inverse multiquadric (IMQ), multilog (ML), multiquadric (MQ), natural cubic spline (NCS) and thin plate spline (TPS). In each case, the shaping parameter R was set to 928m. This is the value obtained from equation (7-50), the default used in Surfer, and it was based on the 3123 control points rather than the entire dataset.

The inverse multiquadric (IMQ) radial basis function was successful. Cholesky decomposition as per sub-appendix G.2.2 was used to solve equation (7-53). Both the latitude revamped signal vector and the longitude revamped signal vector satisfied the solution-check process. Application of (7-51) at the test points produced model signals close to the actual signals. RMS differences were 0.0671m for latitude, 0.0602m for longitude and 0.0902m for horizontal distance.

The multilog (ML) radial basis function was successful. Solution of (7-53) by Cholesky decomposition was impossible because $\mathbf{Rdl}(\mathbf{s},\mathbf{s})$ was not positive definite. (Diagonal elements were smaller than neighbouring elements.) However, the matrix-inversion method of sub-appendix G.2.3 provided solutions for both the latitude revamped signal vector and the longitude revamped signal vector. Each of them satisfied the solution-check process. Application of (7-51) at the test points produced model signals close to the actual signals. RMS differences were 0.0711m for latitude, 0.0579m for longitude and 0.0917m for horizontal distance.

The multiquadric (MQ) radial basis function was successful. Solution of (7-53) by Cholesky decomposition was impossible because $\mathbf{Rdl}(\mathbf{s},\mathbf{s})$ was not positive definite. (Diagonal elements were smaller than neighbouring elements.) However, the matrix-inversion method of sub-appendix G.2.3 provided solutions for both the latitude revamped signal vector and the longitude revamped signal vector. Each of them satisfied the solution-check process. Application of (7-51) at the test points produced model signals close to the actual signals.

RMS differences were 0.0882m for latitude, 0.0633m for longitude and 0.1086m for horizontal distance.

The natural cubic spline (NCS) radial basis function was successful, albeit with an inferior fit to the other RBFs. Solution of (7-53) by Cholesky decomposition was impossible because $\mathbf{Rdl}(\mathbf{s},\mathbf{s})$ was not positive definite. (Diagonal elements were smaller than neighbouring elements.) However, the matrix-inversion method of sub-appendix G.2.3 provided solutions for both the latitude revamped signal vector and the longitude revamped signal vector. Each of them satisfied the solution-check process, to the extent that the largest misclosures were 0.010m for latitude and 0.011m for longitude. Application of (7-51) at the test points produced model signals fairly close to the actual signals. RMS differences were 0.1872m for latitude, 0.1102m for longitude and 0.2172m for horizontal distance.

The thin plate spline (TPS) radial basis function was successful. Solution of (7-53) by Cholesky decomposition was impossible because $\mathbf{Rdl}(\mathbf{s},\mathbf{s})$ was not positive definite. (Diagonal elements were smaller than neighbouring elements.) However, the matrix-inversion method of sub-appendix G.2.3 provided solutions for both the latitude revamped signal vector and the longitude revamped signal vector. Each of them satisfied the solution-check process. Application of (7-51) at the test points produced model signals close to the actual signals. RMS differences were 0.1234m for latitude, 0.0766m for longitude and 0.1453m for horizontal distance.

Inverse distance to a power, described in Section 7.5, was the first of the strictly-bounded SFTs to be applied. It was applied with each of the powers 1, 2, 3 and 4. The predicted signals at the test points became closer to the actual signals as the power n increased. The RMS differences were as follows:

- n=1: 0.1111m for latitude, 0.1512m for longitude, 0.1877m for horizontal distance;
- n=2: 0.0796m for latitude, 0.0851m for longitude, 0.1165m for horizontal distance;
- n=3: 0.0666m for latitude, 0.0608m for longitude, 0.0902m for horizontal distance;
- n=4: 0.0642m for latitude, 0.0586m for longitude, 0.0869m for horizontal distance.

The next form of surface fitting to be applied was HIPFEAD as defined in Section 7.13. Given that estimates of correlation length L had been obtained for the signals for least-squares collocation, $2L$ seemed a reasonable choice for the limit-of-influence r_{max}. This was 28240m

for latitude and 25556m for longitude. The two versions of HIPFEAD are HISFEAD and HICFEAD, described in subsections 7.13.1 and 7.13.2 respectively.

For HISFEAD, the predicted signals at the test points were 20% closer than inverse square distance to the actual signals. The RMS differences were 0.0711m for latitude, 0.0613m for longitude, 0.0938m for horizontal distance.

For HICFEAD, the predicted signals at the test points were 3% closer than inverse cubic distance to the actual signals. The RMS differences were 0.0655m for latitude, 0.0580m for longitude, 0.0874m for horizontal distance.

The next form of surface fitting to be applied was nearest neighbour as described in Section 7.7. One factor in its favour is the large number of control points.

Putting aside (for the moment) the possibility of gridding, nearest neighbour was applied initially to the test points. Comparing the interpolated shifts at the test points to the actual signals, the RMS differences were 0.0751m for latitude, 0.0689m for longitude, 0.1019m for horizontal distance. The interpolating surface, of course, is neither continuous or smooth.

To apply nearest neighbour as a gridding method, datum shifts were predicted at 12160 points, spaced 0.02°(or 72″) apart. These were interpolated at both the test points and the control points. The RMS differences (between actual shifts and interpolated shifts) at the test points were as follows.

- Bilinear interpolation: 0.0711m for latitude, 0.0670m for longitude, 0.0977m for horizontal distance.
- Bicubic interpolation: 0.0743m for latitude, 0.0703m for longitude, 0.1023m for horizontal distance.

The corresponding RMS differences at the control points were roughly 70% smaller (0.0207m, 0.0215m, 0.0299m; 0.0212m, 0.0217m, 0.0303m).

The next form of surface fitting to be applied was LIVONN as defined in Section 7.14. The chosen transition intervals were 33.3333% and 50%. The former interval matches the one illustrated in Figure 7-17.

Putting aside (for the moment) the possibility of gridding, LIVONN was applied initially to the test points. Comparing the predicted signals at the test points to the actual signals, the RMS differences were

- transition interval 33.3333%: 0.0689m for latitude, 0.0649m for longitude, 0.0947m for horizontal distance;
- transition interval 50%: 0.0687m for latitude, 0.0645m for longitude, 0.0942m for horizontal distance.

The interpolating surfaces are not guaranteed to be continuous and are certainly not smooth.

To apply LIVONN with transition interval 33.3333% as a gridding method, datum shifts were predicted at 12160 points, spaced 0.02° (or 72″) apart. These were interpolated at both the test points and the control points. The RMS differences (between actual shifts and interpolated shifts) at the test points were as follows.

- Bilinear interpolation: 0.0664m for latitude, 0.0610m for longitude, 0.0902m for horizontal distance.
- Bicubic interpolation: 0.0680m for latitude, 0.0628m for longitude, 0.0926m for horizontal distance.

The corresponding RMS differences at the control points were roughly 69% smaller (0.0203m, 0.0208m, 0.0291m; 0.0198m, 0.0201m, 0.0283m).

To apply LIVONN with transition interval 50% as a gridding method, datum shifts were predicted at the same 12160 points. These were interpolated at both the test points and the control points. The RMS differences (between actual shifts and interpolated shifts) at the test points were as follows.

- Bilinear interpolation: 0.0663m for latitude, 0.0607m for longitude, 0.0899m for horizontal distance.
- Bicubic interpolation: 0.0678m for latitude, 0.0624m for longitude, 0.0922m for horizontal distance.

The corresponding RMS differences at the control points were roughly 68% smaller (0.0205m, 0.0211m, 0.0294m; 0.0199m, 0.0203m, 0.0284m).

Viewing composite methods as a whole, accuracy comparisons based on test points are given in Table 12-3.

Table 12-3: Accuracy comparison for composite methods in Slovenia, based on residuals at the test points

Surface-Fitting Technique (with parameters)	Horizontal RMS Residual for test points
Inverse Distance to a Power (4)	0.0869
HICFEAD (28240, 25556)	0.0874
LIVONN (50) with BL on grid	0.0899†
Inverse Distance to a Power (3)	0.0902
LIVONN (33.3333) with BL on grid	0.0902†
Radial Basis Function (IMQ, 928)	0.0902
Radial Basis Function (ML, 928)	0.0917
LIVONN (50) with BC on grid	0.0922†
LIVONN (33.3333) with BC on grid	0.0926†
HISFEAD (28240, 25556)	0.0938
LIVONN (50)	0.0942
LIVONN (33.3333)	0.0947
Nearest Neighbour with BL on grid	0.0977†
Nearest Neighbour	0.1019
Nearest Neighbour with BC on grid	0.1023†
Radial Basis Function (MQ, 928)	0.1086
Inverse Distance to a Power (2)	0.1165
Least-Squares Collocation (SS30, 37682, 34102)	0.1416
Radial Basis Function (TPS, 928)	0.1453
Inverse Distance to a Power (1)	0.1877
Radial Basis Function (NCS, 928)	0.2172
Least-Squares Collocation (Gauss, 14120, 12778)	(failed)
Least-Squares Collocation (SS20, 39095, 35379)	(failed)

† Surface not exact at the control points

Some conclusions can be drawn.

- The comparisons of SFTs should be treated with caution. The LIVONN applications with bilinear interpolation from gridded points rank 3rd and 5th in Table 12-3, but they are not smooth. They are also among the daggered cases where the predicted surface is not exact at the control points.
- The dividing line in Table 12-3 separates those methods with a smaller horizontal RMS residual at the test points than the recommended pair of MREs (12-3) and (12-4), which is 0.1193m, from the less accurate methods. By that criteria, even the successful implementation of least-squares collocation would be a poor choice, as would the radial basis functions based on TPS or NCS.
- There is a wide variety of surface-fitting techniques that give a lower horizontal RMS residual than the recommended MREs, although the biggest reduction is a comparatively modest 27%.

This chapter covers the derivation of datum transformations from European Terrestrial Reference System 1989 (ETRS89) to Ordnance Survey Great Britain 1936 (OSGB36). The area of application is Great Britain and there are 4315 data points known in both datums.

4269 of the data points are based on observations. Their distribution was analysed computationally for this study and is illustrated in Figure 13-1; numbers indicate the number of points in each 50km square. The remaining 46 points, along an enclosing boundary, were generated using a 7-parameter transformation. They are shown in Figure 13-2. More details of the datasets are given in sub-appendix C.6.

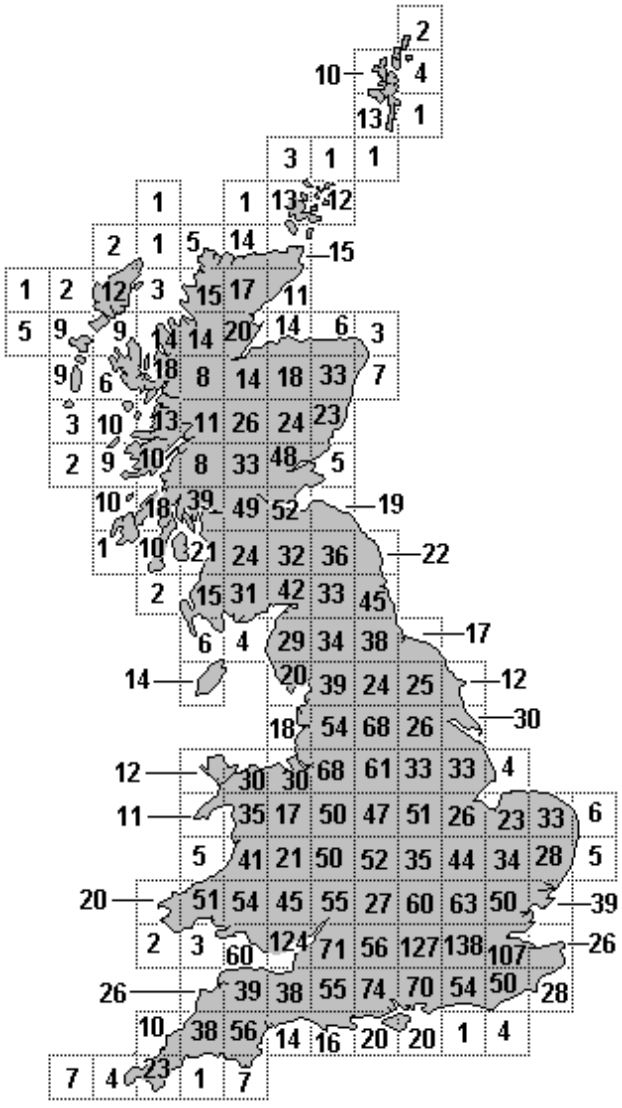


Figure 13-1: Distribution of non-boundary data points for Great Britain, in terms of numbers per 50km-square.

Figure 13-2: Boundary points of the Great Britain dataset, at 50km-intervals on the enclosing rectangle.

The points actually comprise the OSTN15 dataset for which rubber sheeting generated a regular grid for piecewise bilinear interpolation as described in Section 7.3. The ETRS89 coordinates are GNSS-derived. The distortions in OSGB36, however, are such that "different transformations are needed in different parts of the country" (Ordnance Survey [2018], Section 6.3).

As converted for use in this study, the coordinates are (ϕ, λ) in the respective datums. Given the two-dimensional nature of the dataset and the OSGB36 distortions, the types of transformation considered were multiple regression equations and composite methods.

In both cases the dataset was split into:

- 4011 control points from which the transformations were derived, and
- 304 test points to be used for an independent check on accuracy.

Of the 4011 control points, 46 are the boundary points shown in Figure 13-2. Where it seemed appropriate, control points were limited to the 3965 non-boundary control points.

For this study, the algorithm to select test points from the data points was as follows. Grid points were created, 36km apart, around the non-boundary data points. A 26km-by-26km square was put round each grid point, with at least 10km between any two squares. Where there was more than one data point within a square, the 4th, 10th and 20th data points were selected as test points. This is illustrated in Figure 13-3.

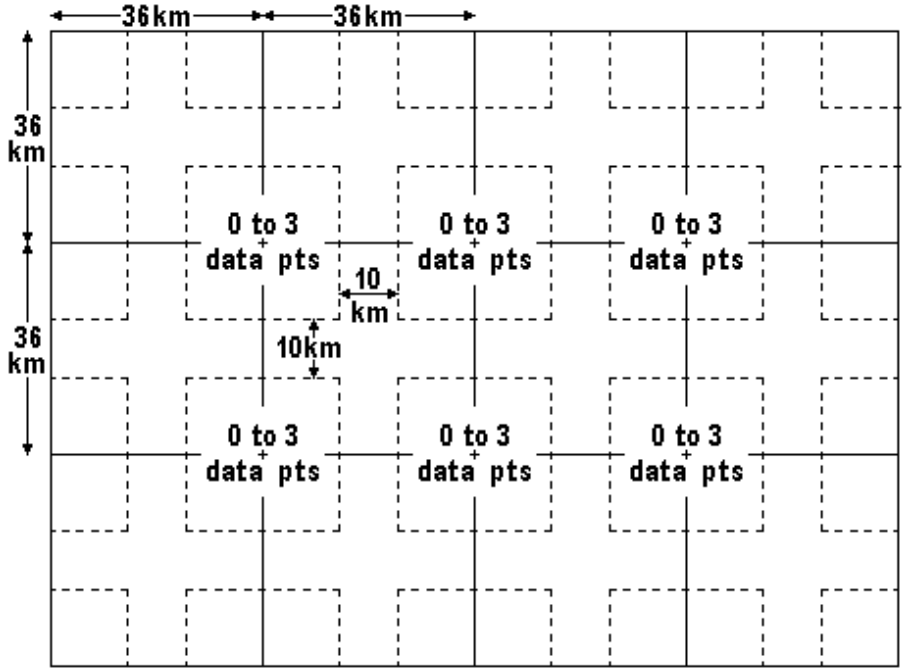


Figure 13-3: Selection of test points from the data points in the dotted 26km squares.

The selection process was designed to ensure that the test points are well spread out and taken from areas with data points to spare. The 10km "corridors" reduces the likelihood of test points being close together although there may be the occasional occurrence within a 26km square containing 10 or more data points.

The test points come from the squares shown in Figure 13-1, which means that none of them is one of the 46 boundary points shown in Figure 13-2 and none of them comes from the sea areas around British shores. The accuracy of a datum transformation model as measured at the test points can only be realistically quoted in relation to the land areas of Great Britain.

One characteristic of the control points, which was not noticed immediately, is the large number of points which are within 50 metres of at least one other point. The most extreme examples are shown in Table 13-1. Interestingly, the official identification numbers of the points within each cluster are very close to each other.

Table 13-1: Control points that are closest to other control points.

Small clusters	Latitude in ETRS89 (deg)	Longitude in ETRS89 (deg)	Residual part of $\Delta\phi$	Residual part of $\Delta\lambda$	Distance apart on ETRS89
Point_0400	57.67890207	-4.57926267	1.10004m	0.53122m	0.039m
Point_0399	57.67890191	-4.57926326	1.11742m	0.56661m	(OSGB36 coords are identical)
Point_0349	50.48613623	-4.69495494	1.44530m	1.57929m	0.051m
Point_0350	50.48613588	-4.69495447	1.48204m	1.54192m	
Point_0156	57.36968677	-5.56449437	0.65113m	-0.03565m	0.085m
Point_0155	57.36968613	-5.56449513	0.72269m	0.00969m	(OSGB36 coords are identical)
Point_4256	52.25765036	1.62865445	-1.92066m	-0.46955m	5.748m
Point_4255	52.25764992	1.62857027	-1.84037m	-0.39127m	
Point_2860	53.26381097	-1.34935750	-1.71029m	-0.11359m	7.205m
Point_2861	53.26374881	-1.34932725	-1.68476m	-0.03427m	
Point_3281	51.42009664	-0.77328510	0.70836m	0.16747m	Between 13.892m and
Point_3280	51.41997273	-0.77330986	0.70773m	0.16865m	16.623m
Point_3279	51.42003083	-0.77349966	0.72744m	0.17417m	
Point_2569	53.75647848	-1.73243349	0.97219m	0.28054m	20.536m
Point_2568	53.75658799	-1.73268410	0.98395m	0.28379m	

13.1 Application of multiple regression equations

The multiple regression equations considered were the fully-normalised MREs described in Section 2.17. That is to say that the intermediate coordinates U and V were defined in such a way that they varied between -1 and 1 over Great Britain.

Only selected MREs are included in this Section. The full list of MREs can be found in sub-appendix H.2.

The offset coordinates in degrees were 56.104 for latitude and -2.968 for longitude. The relative latitude and relative longitude were scaled as follows:

$$U = 0.14662\left(\phi_{\text{in deg}} - 56.104\right); \tag{13-1}$$

$$V = 0.12769\left(\lambda_{\text{in deg}} + 2.968\right). \tag{13-2}$$

The relationship between the geodetic and intermediate coordinates is illustrated in Figure 13-4. The area for which the intermediate coordinates were in the range 1 to -1 was chosen so as to include all the boundary points in Figure 13-2. The reason for this is that the boundary points were included among the control points used for deriving MREs, and were an example of what Section 4.13 calls artificial control points. In retrospect, perhaps the Ordnance Survey boundary points in Figure 13-2 should have been replaced by less remote boundary points (at which datum shifts would have been computed in a similar way). This could have enabled the *V*-axis to divide the land areas more equally.

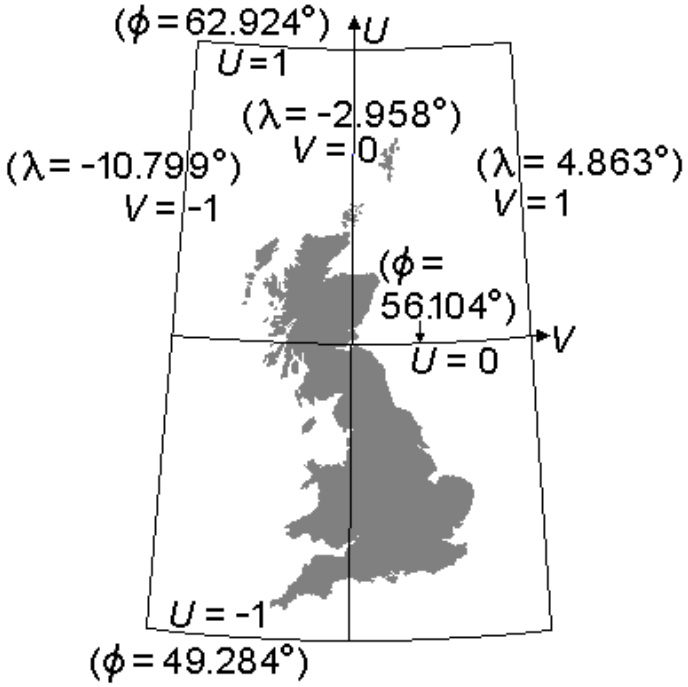


Figure 13-4: The relationship between (ϕ, λ) and (U, V).

The number of points in the dataset meant there was no need for the limit of 40 terms per MRE imposed on Western Australia. However, to keep the size of the expressions manageable, it was decided to set a limit of roughly 50 terms. Taking account of Tables 2-4 to 2-8, this meant setting upper limits on the top power as follows:

- 6 for Ordinary MREs and Chebyshev MREs;
- 4 for North/South MREs, East/West MREs and Four-Quadrant MREs.

The recognised approach for deriving MREs is to apply least-squares optimisation on different combination of terms and to retain those terms which are statistically significant. The

implementation of that approach for this study is the one described in Section 5.5: "eliminating ratios less than one" (ERLTO), where the term "ratio" is the absolute value of a parameter divided by its standard error (AP/SE). The full list of actual MREs is given in sub-appendix H.3.

One characteristic of the derived MREs was that the RMSs of the residuals were smaller at the test points than at the control points. This was at least in part due to the inclusion of all the remote boundary points among the control points.

The Ordinary MREs with top power 3 obtained by ERLTO had 16 terms for the latitude shift and (purely by coincidence) 16 for the longitude shift. The RMSs of the residuals at the test points are 0.3331m for latitude, 0.3374m for longitude and 0.4742m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.4260m, 0.3834m and 0.5732m.)

The Ordinary MREs with top power 4 obtained by ERLTO had 23 terms for the latitude shift and 24 for the longitude shift. The RMSs of the residuals at the test points are 0.3006m for latitude, 0.1834m for longitude and 0.3521m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.3753m, 0.2180m and 0.4340m.)

The Ordinary MREs with top power 5 obtained by ERLTO had 35 terms for the latitude shift and 36 for the longitude shift. The RMSs of the residuals at the test points are 0.1503m for latitude, 0.1722m for longitude and 0.2286m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.1838m, 0.1913m and 0.2653m.)

The Ordinary MREs with top power 6 obtained by ERLTO had 45 terms for the latitude shift and 46 for the longitude shift. The RMSs of the residuals at the test points are 0.1287m for latitude, 0.1364m for longitude and 0.1875m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.1594m, 0.1538m and 0.2215m.)

The North/South MREs with top power 3 obtained by ERLTO had 24 terms for the latitude shift and 22 for the longitude shift. The RMSs of the residuals at the test points are 0.2306m for latitude, 0.1630m for longitude and 0.2824m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.2946m, 0.2065m and 0.3598m.)

Compared with Ordinary MREs with top power 4, the accuracy is 19.8% better and was achieved with 1 fewer term. As it is the most economical MRE-pairing delivering test-point accuracy comparable with 0.2824m, the actual equations are shown here:

$$\Delta\phi('') = \begin{cases} 0.28538 - 0.41802V - 0.11961V^2 - 0.10169V^3 + \\ 3.28408U + 0.27681UV + 0.57604UV^2 - 1.23140UV^3 \\ -0.98424U^2 - 1.45485U^2V - 1.52211U^2V^2 + \\ 3.95979U^2V^3 + 0.57734U^3 + 0.98805U^3V + \\ 0.95184U^3V^2 - 2.65510U^3V^3 \end{cases} \text{ if } U>0$$

$$\begin{cases} +0.37799U^2 + 0.87948U^2V + 1.71478U^2V^2 - \\ 3.07516U^2V^3 - 0.19022U^3 + 0.42040U^3V + \\ 1.51362U^3V^2 - 1.80253U^3V^3 \end{cases} \text{ if } U\leq 0 \qquad (13\text{-}3)$$

$$\Delta\lambda('') = \begin{cases} 5.26013 + 3.90932V - 0.18460V^2 - 0.07808V^3 + \\ 1.06886U + 0.61534UV^2 \\ +0.64301U^2 - 0.84033U^2V - 0.94478U^2V^2 + \\ 1.05226U^2V^3 - 0.30653U^3 + 1.40503U^3V + \\ 0.33394U^3V^2 - 0.98447U^3V^3 \end{cases} \text{ if } U>0$$

$$\begin{cases} +0.09984U^2 - 5.12664U^2V + 2.85239U^2V^2 + \\ 1.76486U^2V^3 - 0.15003U^3 - 4.20115U^3V + \\ 1.94654U^3V^2 + 1.97842U^3V^3 \end{cases} \text{ if } U\leq 0 \qquad (13\text{-}4)$$

The North/South MREs with top power 4 obtained by ERLTO had 40 terms for the latitude shift and 32 for the longitude shift. The RMSs of the residuals at the test points are 0.1485m for latitude, 0.1427m for longitude and 0.2060m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.1735m, 0.1574m and 0.2342m.)

Compared with Ordinary MREs with top power 5, the accuracy is 9.9% better although it required 1 more term.

The East/West MREs with top power 3 obtained by ERLTO had 24 terms for the latitude shift and (purely by coincidence) 24 for the longitude shift. The RMSs of the residuals at the test points are 0.2730m for latitude, 0.2970m for longitude and 0.4034m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.3704m, 0.3429m and 0.5048m.)

Compared with Ordinary MREs with top power 4, the accuracy is 14.6% worse and required 1 more term.

The East/West MREs with top power 4 obtained by ERLTO had 39 terms for the latitude shift and 37 for the longitude shift. The RMSs of the residuals at the test points are 0.2482m for

latitude, 0.1707m for longitude and 0.3012m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.3116m, 0.1997m and 0.3701m.)

Compared with Ordinary MREs with top power 4, the accuracy is 31.8% worse and required 5 more terms.

The Four-Quadrant MREs with top power 3 obtained by ERLTO had 27 terms for the latitude shift and 26 for the longitude shift. The RMSs of the residuals at the test points are 0.2620m for latitude, 0.2029m for longitude and 0.3314m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.3510m, 0.2322m and 0.4209m.)

Compared with Ordinary MREs with top power 4, the accuracy is 5.9% better but was achieved with 6 more terms.

The Four-Quadrant MREs with top power 4 obtained by ERLTO had 51 terms for the latitude shift and 49 for the longitude shift. The RMSs of the residuals at the test points are 0.1786m for latitude, 0.1461m for longitude and 0.2307m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.2242m, 0.1576m and 0.2740m.)

Compared with Ordinary MREs with top power 6, the accuracy is 23.0% worse despite 9 more terms.

The Chebyshev MREs with top power 3 obtained by ERLTO had 16 terms for the latitude shift and (purely by coincidence) 16 for the longitude shift. The RMSs of the residuals at the test points are 0.3331m for latitude, 0.3378m for longitude and 0.4744m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.4260m, 0.3834m and 0.5732m.)

Compared with Ordinary MREs with top power 3, the accuracy is much the same (just 0.05% worse) and involves the same number of terms.

The Chebyshev MREs with top power 4 obtained by ERLTO had 23 terms for the latitude shift and (purely by coincidence) 23 for the longitude shift. The RMSs of the residuals at the test points are 0.3002m for latitude, 0.1837m for longitude and 0.3519m for horizontal

distance. (The corresponding RMS differences at the control points were larger: 0.3753m, 0.2181m and 0.4340m.)

Compared with Ordinary MREs with top power 4, the accuracy is much the same (0.06% better) and requires 1 fewer term.

The Chebyshev MREs with top power 5 obtained by ERLTO had 34 terms for the latitude shift and 36 for the longitude shift. The RMSs of the residuals at the test points are 0.1502m for latitude, 0.1722m for longitude and 0.2285m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.1838m, 0.1913m and 0.2653m.)

Compared with Ordinary MREs with top power 5, the accuracy is same (0.05% better) and requires 1 fewer term.

The Chebyshev MREs with top power 6 obtained by ERLTO had 45 terms for the latitude shift and (purely by coincidence) 45 for the longitude shift. The RMSs of the residuals at the test points are 0.1285m for latitude, 0.1361m for longitude and 0.1872m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.1594m, 0.1538m and 0.2215m.)

Compared with Ordinary MREs with top power 6, the accuracy is much the same (0.2% better) and requires 1 fewer term.

Of the ordinary MREs obtained, the most accurate was the pair with top power 6, denoted "Ord6" for convenience. Of the piecewise MREs obtained (N/S, E/W, 4Q), the most accurate was the N/S pair with top power 4, denoted "NS4" for convenience. For Ord6, the RMS of the test-point residuals was 0.1875m for horizontal distance (from 45+46 terms). For NS4, the RMS of the test-point residuals was 0.2060m for horizontal distance (from 40+32 terms). The superior accuracy of Ord6 is 9.0%, but is achieved by 17 more terms.

To simplify comparisons between Ord6 and NS4, additional trimming of the models (as described in Section 5.5) was applied. The least significant terms were discarded from Ord6(Lat) and Ord6(Lon) until there were 40 and 32 terms respectively. For the trimmed Ord6, denoted "Ord6tr", the RMSs of the residuals at the test points are 0.1310m for latitude, 0.1410m for longitude and 0.1925m for horizontal distance. (The corresponding RMS

differences at the control points were larger: 0.1604m, 0.1587m and 0.2256m.) The accuracy of Ord6tr is 6.6% better than that of NS4, and is achieved with the same number of terms (40+32).

This, however, is not the end of the comparison between Ordinary and North/South MREs. After a North/South MRE with top power 5 was derived by ERLTO, additional trimming of the models was applied until there were 40 terms for latitude and 32 terms for longitude. For the trimmed NS5, denoted "NS5tr", the RMSs of the residuals at the test points are 0.1289m for latitude, 0.1379m for longitude and 0.1887m for horizontal distance. (The corresponding RMS differences at the control points were larger: 0.1507m, 0.1590m and 0.2190m.) The accuracy of NS5tr is 2.0% better than that of Ord6tr, and is achieved with the same number of terms (40+32).

The actual trimmed North/South MREs with top power 5 are

$$\Delta\phi('') = \begin{cases} 0.28274 - 0.38567V + 0.13378V^2 - 0.36550V^4 - 0.06596V^5 + 3.22429U - 1.44981UV^3 + 0.42571UV^4 \\ -0.65998U^2 - 2.42930U^2V^2 + 3.98034U^2V^3 - 3.87840U^3V + 9.04418U^3V^2 - 2.52425U^3V^3 + \\ 0.31318U^4 + 7.87398U^4V - 13.01017U^4V^2 - 4.24886U^5V + 6.12799U^5V^2 & \text{if } U>0 \\ +0.63970U^2 - 24.86874U^2V^2 - 10.12025U^2V^3 + 44.34803U^2V^4 + 2.34166U^3 - 88.46987U^3V^2 - \\ 32.25022U^3V^3 + 150.72671U^3V^4 + 20.59973U^3V^5 + 5.00843U^4 + 0.73673U^4V - 108.58055U^4V^2 - \\ 41.89887U^4V^3 + 176.54452U^4V^4 + 44.03018U^4V^5 + 2.94738U^5 + 0.77772U^5V - 44.63144U^5V^2 - \\ 18.45018U^5V^3 + 69.22826U^5V^4 + 23.39180U^5V^5 & \text{if } U\leq 0 \end{cases} \quad (13\text{-}5)$$

$$\Delta\lambda('') = \begin{cases} 5.26546 + 3.91765V - 0.19198V^2 + 1.10162U + 0.32742UV + 0.84945UV^2 - 0.76076UV^3 - 0.8207UV^4 \\ -3.09588U^2V + 2.91768U^2V^3 + 0.95188U^2V^4 + 3.35184U^3V + 3.11526U^4 - 5.09247U^4V^2 - \\ 2.25876U^4V^3 - 2.83209U^5 + 4.17931U^5V^2 & \text{if } U>0 \\ +9.59989U^2V^2 - 12.38098U^2V^3 - 3.79466U^2V^4 - 0.59667U^3 + 15.63958U^3V + 26.3831U^3V^2 - \\ 42.87075U^3V^3 - 3.29270U^3V^4 - 0.35837U^4 + 28.33820U^4V + 31.84657U^4V^2 - 51.55637U^4V^3 + \\ 13.70320U^5V + 14.12399U^5V^2 - 20.60202U^5V^3 & \text{if } U\leq 0 \end{cases} \quad (13\text{-}6)$$

This comparison process could be continued indefinitely with Ordinary and North/South MREs leapfrogging each other by including a wider range of monomials to choose from. The justification for stopping it at NS5tr is the desirability of avoiding high-power polynomials.

The presence of large coefficients in (13-5) and (13-6) might appear to be a cause for concern, although the discussion of Chebyshev MREs below addresses this point. In addition, there is the evidence of the test-point accuracy. The test points come from the squares shown in Figure 13-1, and the vast majority of data points from those squares provided control. As mentioned earlier, test-point accuracy is only an indication of accuracy over the land areas.

The Chebyshev MREs in sub-appendix H.3 show a distinct tendency to have smaller coefficients than the MREs based on monomials. The significance of this can be deduced by comparing the initial ordinary MRE with top power 6 with the initial Chebyshev MRE with top 6; "initial" means "prior to application of ERLTO" so that they are different representations of the same polynomial. The L_1 norms of the coefficients were as follows:

- for the latitude shift, 217.1634 (Ord) and 6.2097 (Cheb);
- for the longitude shift, 226.0569 (Ord) and 14.3625 (Cheb).

Equivalence of the functions means that the lower Chebyshev L_1 norms are limits on the magnitudes of the ordinary MREs. (See subsections 2.17.1 and 2.17.5.) This implies that some of the seemingly-large monomial terms largely cancel each other out. It is logical to assume that this is true of the monomial-based MREs in general, including N/S, E/W and 4Q. This revelation is perhaps the most significant contribution of Chebyshev polynomials to the study of MREs in Great Britain.

Contour maps of most of the MREs are shown in in sub-appendix I.3. They show strong similarities to one another, with no sharp bends or major variations in spacing.

The main finding of this section is that North/South MREs work well for transforming ETRS89 to OSGB36. The North/South MREs in (13-3) and (13-4) have a greater emphasis on economy than the more accurate North/South MREs of !3-5) and (13-6), but in both cases they compare favourably with ordinary MREs with the same number of terms.

13.2 Application of composite methods

The composite methods applied in Great Britain were selected from those described in Chapter 7 and the methods for interpolation of gridded data were selected from those described in Chapter 6. The generic process used is illustrated in Figure 13-5.

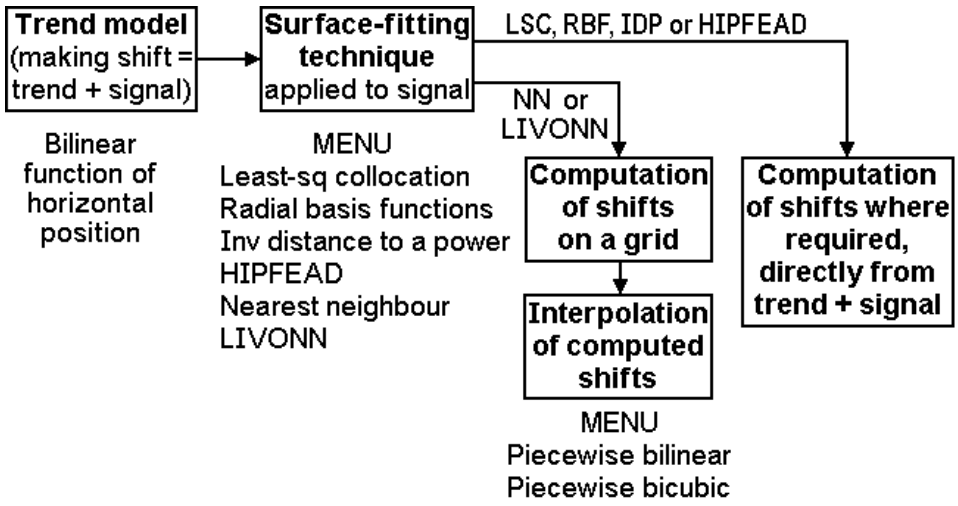


Figure 13-5: Composite methods chosen for ETRS89→OSGB36 in Great Britain.

The trend model chosen for the latitude shift (in arc-seconds) was

$$\Delta\phi_{\mathrm{BLM}}('') = 0.25288 + 0.425846\left(\phi_{\mathrm{in\ deg}} - 56.104\right) - 0.069947\left(\lambda_{\mathrm{in\ deg}} + 2.968\right) - 0.0030543\left(\phi_{\mathrm{in\ deg}} - 56.104\right)\left(\lambda_{\mathrm{in\ deg}} + 2.968\right). \quad (13\text{-}7)$$

The trend model chosen for the longitude shift (in arc-seconds) was

$$\Delta\lambda_{\mathrm{BLM}}('') = 5.27689 + 0.143205\left(\phi_{\mathrm{in\ deg}} - 56.104\right) + 0.46703\left(\lambda_{\mathrm{in\ deg}} + 2.968\right) + 0.018828\left(\phi_{\mathrm{in\ deg}} - 56.104\right)\left(\lambda_{\mathrm{in\ deg}} + 2.968\right). \quad (13\text{-}8)$$

In each case, the trend model is the bilinear model that gives the least-squares fit to the shifts at the control points, hence the subscript BLM. This model is mathematically equivalent to the multiple regression equation with top power 1. The bilinear form has sufficient terms to ensure that

- the trend model includes the average shift as a constant term, and
- embodies the tilt which is evident from the contour maps I-43 to I-51 in Appendix I.

For both the latitude shift and the longitude shift, the signal component is that part which is unmodelled.

The surface-fitting methods tested for each signal were those shown in Figure 13-5. They are also listed below with the corresponding Sections.

- Least-squares collocation (7.1);
- Radial basis functions (7.6);
- Inverse distance to a power (7.5);
- Hybrid inverse power function embodying accelerated decline [HIPFEAD] (7.13);
- Nearest neighbour (7.7);
- Linear interpolation variation on nearest neighbour [LIVONN] (7.14).

Figure 13-5 differs from Figure 8-3 in that the emphasis has shifted from SFTs as a gridding method to SFTs as an approximation method to be applied wherever required. The first four techniques generate a C^1 surface and do not require a huge amount of computation. It is true that least-squares collocation and radial basis functions require a one-off computation of revamped signals from a large matrix equation. However, this would be the case if the methods were employed as a gridding method. The difference is that direct application requires storage of the revamped signals at control points instead of the signal values at grid points. The direct approach has the advantage that it interpolates exact data values rather than approximate values at grid points.

The SFTs listed which do not generate a C^1 surface are nearest neighbour and LIVONN. Nearest neighbour creates discontinuities and LIVONN is liable to do so in places. These were treated as gridding methods to be followed by bilinear interpolation (Section 6.2) or bicubic interpolation (6.3, as in 6.3.3). The grid was defined by intervals of 6′ (0.1°).

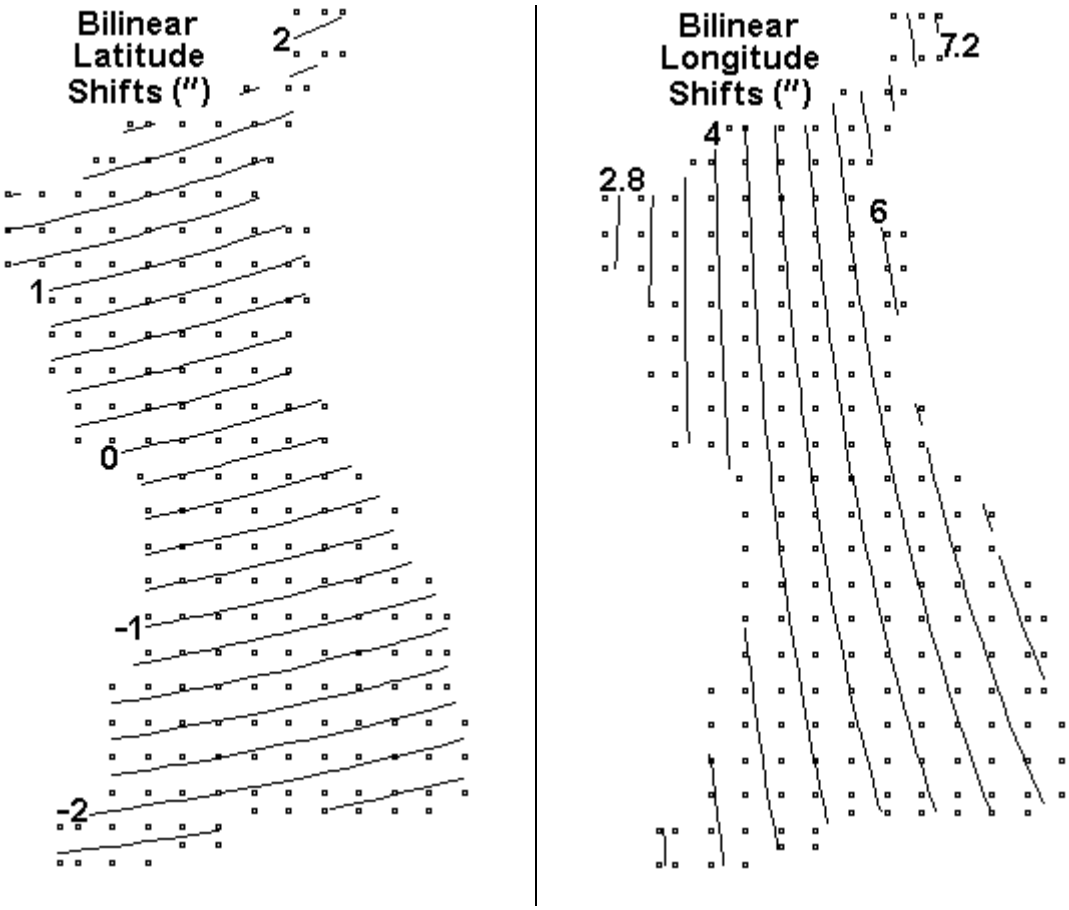


Figure 13-6: Contour maps of the trend models of ETRS89→OSGB36 in arc-seconds.

Figure 13-6 shows the contours generated by LSS for the trend model defined by (13-7) and (13-8). (The points used in these maps are considerably fewer than the 6′ grid of computation points.)

Least-squares collocation was applied with three covariance functions applied to the signals. These were Gaussian, SS20 and SS30, described in subsection 7.1. Variance (in square metres) was 1.421349 for the latitude signal and 0.554053 for the longitude signal. In both cases this was computed from the non-boundary control points.

The size of the dataset made it possible to estimate the correlation length (and hence the 2nd constant required by each covariance function) by sampling at the non-boundary control points. Tables 13-2 and 13-3 feature the ranges of distance for which the estimated correlation was close to 0.5.

Table 13-2: Sampling of latitude-signal products to estimate covariance and correlation

Min of Range	Max of Range	Centre of Range	Sample Size	Estimated Covariance From Mean	Estimated Correlation From Mean
94000	95000	94500	14249	0.728444	0.512502
95000	96000	95500	14253	0.709402	0.499105
96000	97000	96500	14040	0.710339	0.499764
97000	98000	97500	13999	0.710029	0.499546
98000	99000	98500	14284	0.700164	0.492605

The linear function giving the least-squares fit to the data in columns 3 and 6 is $\text{Correlation} = 1.506795 - 0.0000104258 \times \text{Centre}$. Substitution of 0.5 for Correlation gives a value of 96567.56 for Centre, so the estimated correlation length for the latitude signal is 96568m.

Table 13-3: Sampling of longitude-signal products to estimate covariance and correlation

Min of Range	Max of Range	Centre of Range	Sample Size	Estimated Covariance From Mean	Estimated Correlation From Mean
13000	14000	13500	3347	0.293953	0.530551
14000	15000	14500	3599	0.280526	0.506317
15000	16000	15500	3778	0.267881	0.483494
16000	17000	16500	4064	0.286327	0.516786
17000	18000	17500	4093	0.261357	0.471718
18000	19000	18500	4322	0.277808	0.501411
19000	20000	19500	4399	0.276172	0.498459
20000	21000	20500	4587	0.277445	0.500756
21000	22000	21500	4749	0.263834	0.476189
22000	23000	22500	5076	0.270683	0.488552

The linear function giving the least-squares fit to the data in columns 3 and 6 is $\text{Correlation} = 0.516275 - 0.0000010473 \times \text{Centre}$. Substitution of 0.5 for Correlation gives a value of 15539.68 for Centre, so the estimated correlation length for the longitude signal is 15540m.

The conversion of correlation length into the covariance function's 2nd constant (k or r_{max}) is described in Examples 7.1 to 7.3 in Chapter 7. The covariance functions to be adopted were therefore as follows:

- Gaussian, as per formula (7-11): for the latitude signal, $C_0 = 1.421349$ and $k = 1/115989$; for the longitude signal, $C_0 = 0.554053$ and $k = 1/18665$.
- SS20 as per formula (7-14): for the latitude signal, $C_0 = 1.421349$ and $r_{max} = 267370$; for the longitude signal, $C_0 = 0.554053$ and $r_{max} = 43025$.
- SS30 as per formula (7-17): for the latitude signal, $C_0 = 1.421349$ and $r_{max} = 257711$; for the longitude signal, $C_0 = 0.554053$ and $r_{max} = 41471$.

Initially, least-squares collocation was applied with a "no noise" assumption. For each of the covariance functions, the revamped signal was successfully computed. For the Gaussian function, the Cholesky decomposition as per sub-appendix G.2.2 broke down in row 1502 for longitude; for SS20, Cholesky broke down in row 52 for latitude and row 684 for longitude). The matrix-inversion method of sub-appendix G.2.3 was applied successfully where Cholesky failed, although the values of the revamped signals were suspiciously large.

The predicted signals at the test points were another matter. Comparing the predicted signals with the actual signals, the RMS differences were as follows:

- Gaussian: 0.0655m for the latitude signal, 14228.695m for the longitude signal and 14228.695m for the horizontal distance.
- SS20: 1530.2956m for the latitude signal, 12710.6117m for the longitude signal and 12802.4004m for the horizontal distance.
- SS30: 53.9235m for the latitude signal, 80.0258m for the longitude signal and 96.4981m for the horizontal distance.

The reason for these implausible signals lies in the rows of the matrix $\mathbf{Cov}(\mathbf{s}, \mathbf{s})$. Where data points are virtually identical, as per the examples given in Table 13-1, the corresponding rows of $\mathbf{Cov}(\mathbf{s}, \mathbf{s})$ are going to be virtually identical, making the matrix equation (7-29) very ill-conditioned. The distance between points in these clusters is too small for HISFEAD to be used as a modification of least-squares collocation.

Another aspect of Table 13-1 is the difference between residuals at near-identical points. This makes the "no noise" assumption totally unrealistic. In other words, not all of each residual (from the actual shift minus the trend model) can realistically be treated as signal. Some allowance has to be made for noise in the actual datum shifts which will translate into noise in the residuals.

The process described in equations (7-30) to (7-34) of Section 7.1 was applied. The variances of noise for latitude and longitude were assumed to be the same. They were set, in turn, to $0.04m^2$, $0.0016m^2$, $0.0004m^2$ and $0.0001m^2$. The signal covariances for latitude and longitude were reduced accordingly. This had the effect of making the diagonal terms slightly more dominant in the covariance matrix, making equation (7-33) better conditioned. It also produced predicted signals for cluster points that eliminated the differences.

When the Gaussian covariance function was used, least-squares collocation was successful for each of the four signal-noise splits. When the noise variance was set to $0.0001m^2$, the RMS differences between predicted and actual signals at the test points were 0.0655m for the latitude signal, 0.0799m for the longitude signal and 0.1033m for horizontal distance. This was offset by RMS differences of 0.0685m, 0.0293m and 0.0745m at the control points; these were higher than expected given the size of the noise variance.

When the SS20 covariance function was used, least-squares collocation was unsuccessful for each of the four signal-noise splits. The RMS differences between predicted and actual signals at the test points were between 1 and 7 metres for horizontal distance. One indication that the covariance matrix was ill-conditioned was that Cholesky decomposition was only possible when the noise variance was $0.04m^2$.

When the SS30 covariance function was used, least-squares collocation was successful for each of the four signal-noise splits. When the noise variance was set to $0.0001m^2$, the RMS differences between predicted and actual signals at the test points were 0.0527m for the latitude signal, 0.0710m for the longitude signal and 0.0884m for horizontal distance. This was offset by RMS differences of 0.0272m, 0.0085m and 0.0285m at the control points; these were less in conflict with the size of the noise variance than was the case with the Gaussian covariance function.

Radial basis functions were applied using all the examples described in subsection 7.6: inverse multiquadric (IMQ), multilog (ML), multiquadric (MQ), natural cubic spline (NCS) and thin plate spline (TPS). In each case, the shaping parameter R was set to 4021m. This is the value obtained from equation (7-50), the default used in Surfer, and it was based on the 3965 non-boundary control points rather than the entire dataset.

For each of the radial basis functions, the revamped signal vector was successfully computed. Equation (7-53) was solved by Cholesky decomposition as per sub-appendix G.2.2 in the case of inverse multiquadric (IMQ). The matrix-inversion method of sub-appendix G.2.3 was used for the other RBFs. The quality of the revamped signal vectors was in doubt for natural cubic spline (NCS) and thin plate spline (TPS), because the solution-check process produced large misclosures.

The predicted signals at the test points varied from disappointing to unusable, although they were not as bad as noise-free least-squares collocation. Comparing the predicted signals with the actual signals, the RMS differences were as follows:

- Inverse multiquadric (IMQ): 0.2524m for the latitude signal, 0.2559m for the longitude signal and 0.3594m for the horizontal distance.
- Multilog (ML): 0.8053m for the latitude signal, 0.9369m for the longitude signal and 1.2354m for the horizontal distance.
- Multiquadric (MQ): 5.2979m for the latitude signal, 5.3997m for the longitude signal and 7.5647m for the horizontal distance.
- Natural cubic spline (NCS): 61.9467m for the latitude signal, 35.9614m for the longitude signal and 71.6283m for the horizontal distance.
- Thin plate spline (TPS): 24.0903m for the latitude signal, 40.0007m for the longitude signal and 46.6948m for the horizontal distance.

The reason for these results is much the same as the reason for the results from noise-free least-squares collocation. However, radial basis functions have no comparable method for making allowance for noise.

Inverse distance to a power, described in Section 7.5, was the first of the strictly-bounded SFTs to be applied. It was applied with each of the powers 1, 2, 3 and 4. The predicted signals at the test points became closer to the actual signals as the power n increased. The RMS differences were as follows:

- n=1: 0.8517m for latitude, 0.5425m for longitude, 1.0098m for horizontal distance;
- n=2: 0.2471m for latitude, 0.1851m for longitude, 0.3087m for horizontal distance;
- n=3: 0.0678m for latitude, 0.0651m for longitude, 0.0940m for horizontal distance;
- n=4: 0.0598m for latitude, 0.0574m for longitude, 0.0829m for horizontal distance.

The next form of surface fitting to be applied was HIPFEAD as defined in Section 7.13. Given that estimates of correlation length L had been obtained for the signals for least-squares collocation, $2L$ seemed a reasonable choice for the limit-of-influence r_{max}. This was 193136m for latitude and 31080m for longitude. The two versions of HIPFEAD are HISFEAD and HICFEAD, described in subsections 7.13.1 and 7.13.2 respectively.

For HISFEAD, the predicted signals at the test points were 52% closer than inverse square distance to the actual signals. The RMS differences were 0.1363m for latitude, 0.0566m for longitude, 0.1476m for horizontal distance.

For HICFEAD, the predicted signals at the test points were 11% closer than inverse cubic distance to the actual signals. The RMS differences were 0.0629m for latitude, 0.0557m for longitude, 0.0840m for horizontal distance.

The next form of surface fitting to be applied was nearest neighbour as described in Section 7.7. One factor in its favour is the large number of control points.

Putting aside (for the moment) the possibility of gridding, nearest neighbour was applied initially to the test points. Comparing the predicted signals at the test points to the actual signals, the RMS differences were 0.0800m for latitude, 0.0704m for longitude, 0.1066m for horizontal distance. The interpolating surface, of course, is neither continuous or smooth.

To apply nearest neighbour as a gridding method, datum shifts were predicted at 12090 points, spaced 0.1°(or 6′) apart. These were interpolated at both the test points and the non-boundary control points. The RMS differences (between actual shifts and interpolated shifts) at the test points were as follows.

- Bilinear interpolation: 0.0681m for latitude, 0.0642m for longitude, 0.0936m for horizontal distance.
- Bicubic interpolation: 0.0735m for latitude, 0.0685m for longitude, 0.1004m for horizontal distance.

The corresponding RMS differences at the non-boundary control points were roughly 32% smaller (0.0471m, 0.0445m, 0.0648m; 0.0495m, 0.0462m, 0.0677m).

The next form of surface fitting to be applied was LIVONN as defined in Section 7.14. The chosen transition intervals were 33.3333% and 50%. The former interval matches the one illustrated in Figure 7-17.

Putting aside (for the moment) the possibility of gridding, LIVONN was applied initially to the test points. Comparing the predicted signals at the test points to the actual signals, the RMS differences were

- transition interval 33.3333%: 0.0689m for latitude, 0.0613m for longitude, 0.0922m for horizontal distance;
- transition interval 50%: 0.0686m for latitude, 0.0612m for longitude, 0.0919m for horizontal distance.

The interpolating surfaces are not guaranteed to be continuous and are certainly not smooth.

To apply LIVONN with transition interval 33.3333% as a gridding method, datum shifts were predicted at 12090 points, spaced 0.1°(or 6′) apart. These were interpolated at both the test points and the non-boundary control points. The RMS differences (between actual shifts and interpolated shifts) at the test points were as follows.

- Bilinear interpolation: 0.0633m for latitude, 0.0593m for longitude, 0.0867m for horizontal distance.
- Bicubic interpolation: 0.0663m for latitude, 0.0612m for longitude, 0.0903m for horizontal distance.

The corresponding RMS differences at the non-boundary control points were roughly 32% smaller (0.0432m, 0.0416m, 0.0600m; 0.0443m, 0.0421m, 0.0612m).

To apply LIVONN with transition interval 50% as a gridding method, datum shifts were predicted at the same 12090 points. These were interpolated at both the test points and the non-boundary control points. The RMS differences (between actual shifts and interpolated shifts) at the test points were as follows.

- Bilinear interpolation: 0.0662m for latitude, 0.0592m for longitude, 0.0866m for horizontal distance.
- Bicubic interpolation: 0.0662m for latitude, 0.0610m for longitude, 0.0900m for horizontal distance.

The corresponding RMS differences at the non-boundary control points were roughly 32% smaller (0.0431m, 0.0416m, 0.0599m; 0.0441m, 0.0419m, 0.0609m).

Viewing composite methods as a whole, accuracy comparisons based on test points are given in Table 13-4.

Table 13-4: Accuracy comparison for composite methods in Great Britain, based on residuals at the test points

Surface-Fitting Technique (with parameters)	Horizontal RMS Residual for test points
Inverse Distance to a Power (4)	0.0829
HICFEAD (193136, 31080)	0.0840
LIVONN (50) with BL on grid	0.0866†
LIVONN (33.3333) with BL on grid	0.0867†
Least-Squares Collocation (SS30, 257711, 41471)‡	0.0884†
LIVONN (50) with BC on grid	0.0900†
LIVONN (33.3333) with BC on grid	0.0903†
LIVONN (50)	0.0919
LIVONN (33.3333)	0.0922
Nearest Neighbour with BL on grid	0.0936†
Inverse Distance to a Power (3)	0.0940
Nearest Neighbour with BC on grid	0.1004†
Least-Squares Collocation (Gauss, 96568, 15540)‡	0.1033†
Nearest Neighbour	0.1066
HISFEAD (193136, 31080)	0.1476
Inverse Distance to a Power (2)	0.3087
Radial Basis Function (IMQ, 4021)	0.3594
Inverse Distance to a Power (1)	1.0098
Least-Squares Collocation (SS20, 267370, 43025)‡	(failed)
Radial Basis Function (ML, 4021)	(failed)
Radial Basis Function (MQ, 4021)	(failed)
Radial Basis Function (NCS, 4021)	(failed)
Radial Basis Function (TPS, 4021)	(failed)

‡With allowance for noise † Surface not exact at the control points

Some conclusions can be drawn.

- The comparisons of SFTs should be treated with caution. The LIVONN applications with bilinear interpolation from gridded points rank 3rd and 4th in Table 13-4, but they are not smooth. As with least-squares collocation using SS30, they are among the daggered cases where the predicted surface is not exact at the control points.
- The dividing line in Table 13-4 separates those methods with a smaller horizontal RMS residual at the test points than the recommended pair of MREs (13-5) and (13-6), which is 0.1887m, from the less accurate methods. By that criteria, even the successful radial basis function would be a poor choice.
- There is a wide variety of surface-fitting techniques that give a lower horizontal RMS residual than the recommended MREs. Most compare favourably with OSTN15, for which the RMS error is said to be 0.1m (Ordnance Survey [2018, Section 6.3]).

Part Four: Conclusions And References

CHAPTER 14: CONCLUSIONS

There is a wide variety of methods for transforming coordinates between geodetic datums. This is due not only to the large number of functions designed for that purpose, but also the existence of a vast range of interpolation methods classified as surface-fitting techniques.

There is no single transformation method which is superior to all others. The following considerations provide some of the reasons for this.

- The relationship between coordinates in different datums is influenced by differences in the methods of measurement used (Sections 1.6 and 1.7). That varies from case to case.
- In many instances, ellipsoidal heights in either datum are unknown or only known to a high degree of uncertainty, undermining comparisons of 3D Cartesian coordinates. In such cases, methods that just transform latitude & longitude may be the most suitable.
- The choice between transformations that fit data exactly and those that allow for "noise" by smoothing depends on the quality of the coordinates in the given datasets. That in turn depends on the nature and accuracy of the measurements used in each datum (an issue discussed in Chapter 1). Distinguishing between noise and actual distortion isn't always easy. Section 13.2 provided an instance where an assumption of zero noise would have made least-squares collocation untenable.
- Accuracy-of-fit to actual data is sometimes a secondary consideration to ease of computation and application. Sections 2.3 and 2.8 describe how two of the simpler methods are well-suited to software packages that cater for a wide range of datums.

This thesis widens the choice by introducing new transformation methods. The basic ones are SMITSWAM (which is conformal), partially-conformal variations on Standard & Abridged Molodensky (near-conformal horizontally) and normalised generalisations of multiple regression equations (5 types, usually non-conformal). They are described in Sections 2.13 to 2.17, and their properties are summarised in Section 14.1 below. The new methods based on surface-fitting techniques are HIPFEAD and LIVONN. They are described in Sections 7.13 & 7.14, and their properties are summarised in subsection 14.5.2 below.

This study compares the properties of different methods at a theoretical level. Practical comparisons between methods have also been included, using a varied selection of datasets.

At both levels, this thesis provides indications of which methods are best in terms of accuracy, computational convenience and numerical stability, taking into account the characteristics of the common-point datasets and the areas of coverage.

This study also introduces new methods of deriving transformations, summarised in Sections 14.3 & 14.4 below, and new methods of reversing transformations, summarised in Section 14.2.

14.1 Basic transformation methods

The Helmert transformation varies according to the order in which the rotations are applied. Only two of the six possible permutations are used in practice. This study offers a process to convert one version of Helmert to the other (Section 2.6 and Ruffhead [2021b]). This enables published parameters of one version to be used in software designed for the other.

This study appears to be alone in acknowledging that there are two versions of the Bursa-Wolf transformation, *totally-linear* and *partially-linear*, and that there is a simple & exact relationship between the two. The Molodensky-Badekas method is shown to have the same properties. Because Molodensky-Badekas is computationally equivalent to Bursa-Wolf, this thesis firmly supports those researchers who assert that the former has no advantage over Bursa-Wolf.

The popularity of Standard Molodensky and Abridged Molodensky owes much to "Cartesian bypassers". These are scientists & technicians who want to apply near-conformal transformations without converting geodetic coordinates to and from Cartesian coordinates. Without endorsing that preference (which would rule out Helmert and Bursa-Wolf), this researcher has extended what can be achieved within that limitation.

- Standard Molodensky in two stages with applied misclosure (SMITSWAM) makes Standard Molodensky truly conformal. This process (Section 2.14 and Ruffhead [2016]) means that Cartesian bypassers need no longer settle for near-conformal 3-parameter transformations.

- Both Standard and Abridged Molodensky can be varied to partially-conformal models with 6 or 7 parameters. The evidence of Chapters 8 to 11 shows that the fit to data is improved, in some cases substantially. Beyond the derivation of parameters, there is

negligible extra computation. These new variations (Sections 2.15 and 2.16) offer Cartesian bypassers more accurate options, particularly when horizontal & vertical coordinates have been obtained by different methods and/or there is evidence of a *Z*-rotation.

This study examined 5 types of fully-normalised multiple regression equations (Section 2.17), all of them with intermediate coordinates in the range -1 to 1. They are all original methods with the arguable exception of "ordinary" MREs. The latter differ from the type recommended by Appelbaum only in the use of 2 scaling multiples for the intermediate coordinates. The advantage of full normalisation is that coefficients can be computed to the same precision as that intended for the MRE as a summation.

Three types of MRE are recommended for comparison with ordinary MREs when MREs are developed for future datum transformation models. They reduce the need for high-order monomials as basis functions. For each of them, there are instances where it provides a better compromise between accuracy and economy than ordinary MREs. The types are as follows:

- North/South MREs, which were found to have advantages over traditional MREs for transforming ETRS89 to OSGB36 in Great Britain (Section 12.1). The North/South MREs with top power 3 were 20% more accurate than ordinary MREs with a similar number of terms. The advantage, albeit smaller, is retained when higher-power terms are involved. It is significant that the north-south extent greatly exceeds the east-west extent.

- East/West MREs, which were found to have advantages over traditional MREs for transforming D48 to D96 in Slovenia (Section 13.1). In contrast to Great Britain, the east-west extent of Slovenia exceeds the north-south extent. Analysis of results showed that the best compromise between economy and accuracy was a combination of a trimmed Ordinary MRE with top power 6 for the latitude shift and a trimmed East/West MRE with top power 4 for the longitude shift.

- Four-Quadrant MREs, which compared favourably with traditional MREs for transforming AGD84 to GDA94 in Western Australia (Section 8.2). Analysis of results showed that the best compromise between economy and accuracy was a

combination of an Ordinary MRE with top power 4 for the latitude shift and a Four-Quadrant MRE with top power 3 for the longitude shift.

In all three case studies, comparing these types with ordinary MREs resulted in a more economical transformation model than would have been achieved by only considering ordinary MREs.

Chebyshev MREs (subsection 2.17.5) were found to have no practical advantages over ordinary MREs. In general, they have just as many statistical terms and require more processing time. They have theoretical significance, however, in that they show that monomial-based MREs are more stable than might be supposed from the size of the coefficients; in other words, seemingly-large monomial terms largely cancel each other out. From 6 instances of MREs where the Chebyshev MRE was equivalent to the ordinary MRE, the former's L_1 norm of the coefficients was lower by between 54% and 97%.

14.2 Reverse transformations

Published descriptions of reverse transformations (*eg* Iliffe and Lott [2008] and Knippers [2009]) tend to omit rearrangement-type formulae that provide exact inverses. Several methods are reversible by this approach, including the following:

- Conformal transformation in 2 dimensions (Section 3.1);
- Affine transformation in 2 dimensions (Section 3.2);
- Helmert transformation (Section 3.6);
- Localised rigorous 7-parameter conformal transformation (Section 3.7);
- Bursa-Wolf (Section 3.8);
- Molodensky-Badekas (Section 3.9);
- 8-parameter affine transformation (Section 3.10);
- 9-parameter affine transformation (Section 3.11);
- 12-parameter affine transformation (Section 3.12).

Most rearrangement-type formulae can be found individually in publications concentrating on one or two transformation methods. Ruffhead and Whiting (2020) provides comprehensive coverage of their use.

Although Helmert is among the transformations which can be reversed by a rearrangement-type formula, it is recognised that some scientists and technicians prefer a same-formula inverse with different values for the parameters. This need not be the simple same-formula

inverse in which the parameters of the original formula are applied with opposite signs, giving only approximate results. The correct parameters for the same-formula inverse can be derived by applying the procedure for converting one version of Helmert to the other (Section 3.6 and Ruffhead [2021b]).

There are some transformation methods where there is no rearrangement-type formula to compute the inverse, but there is a simple same-formula inverse (SSFI) that gives an approximate result. By applying the original formula to the result of the SSFI, a misclosure is obtained which provides a very effective correction to the SSFI. The methods covered by this approach are

- Standard Molodensky (Section 3.4 and Ruffhead and Whiting [2020]);
- Abridged Molodensky (Section 3.5 and Ruffhead and Whiting [2020]);
- Variations on Standard Molodensky (Section 3.15);
- Variations on Abridged Molodensky (Section 3.16).

14.3 Derivation of transformations

This thesis introduces a new method of deriving 7-parameter conformal (Helmert) models (Section 5.1 and Ruffhead [2021a]). Determining scale change by distance analysis is not unprecedented, but the way it is combined with a two-stage derivation of a rigid transformation is believed to be original. The process can be applied to both versions of Helmert. It is easier to apply than methods cited in Section 5.1 which require knowledge of such advanced concepts as Groebner basis and Procrustean solution.

This thesis also introduces a new method of deriving affine transformations with 8 or 9 parameters (Sections 5.3 and 5.4). The starting point is the optimum 7-parameter model described in Section 5.1. An “equivalent-enlargement hypothesis” is applied to enlarge the set of optimal parameters from 7 to 8, and (if desired) from 8 to 9.

Optimising multiple regression equations is complicated by the fact that the number of parameters is not fixed. The parameters are coefficients of the basis functions, so that number coincides with the number of terms. For a given set of basis functions, the coefficients are obtained by least-squares optimisation. The criterion for inclusion of terms is statistical significance, but there are many ways in which that can be applied. The approach used by NIMA (formerly DMA) involved adding terms to the least-squares process as well as removing them, but DMA (1987a, Section 7.2.4.3.3) is sparse on detail.

In this study, the starting point was a given type of MRE with a given top power. Coefficients were computed by least-squares and their standard errors were deduced from the inverse normal matrix. Where the value of the absolute parameter divided by its standard error was less than 1, *ie* AP/SE<1, the term was a candidate for removal. The approach devised for this thesis was therefore Eliminating Ratios Less Than One (ERLTO); this is described in detail in Section 5.5.

It was discovered that it is usually possible to remove additional terms with a negligible effect on accuracy, particularly when AP/SE is between 1 and 3. That offered scope for what this thesis describes as "Additional Trimming". It was used to enable the comparisons between different types of MRE, which are summarised in Section 14.1.

14.4 Piecewise interpolation-based methods

Gridding for interpolation purposes doesn't necessarily require a *projected* grid. Regular spacing can be done in terms of geodetic coordinates and distances can be lengths of ellipsoidal arcs. The latter can be approximated using the simple equation (7-1) rather than a lengthy algorithm for the geodesic.

There are many methods of piecewise interpolation that can be applied to points enclosed by a regular grid. The two considered in this study are bilinear interpolation and bicubic interpolation. Both generate a continuous surface. The bicubic algorithms in Section 6.3 were derived for this thesis and are believed to be original to the extent that they are based on 12 points rather than 16. They generate a smooth surface and do not require the solution of matrix equations. A choice of 3 algorithms is provided (subsections 6.3.1 to 6.3.3).

In the interpolation of datum shifts, the C^1 property of the piecewise bicubic surface gives it an apparent advantage over the piecewise bilinear surface. However, in the case studies, the bilinear method was more accurate than the bicubic method when applied to data interpolated by nearest neighbour or LIVONN (see subsection 14.5.2).

14.5 Composite methods

These were introduced in Chapter 7. In this thesis a composite method is one that treats a datum shift as either

- the sum of a trend model and signal, or

- interpolation of a regular grid of predicted shifts that were obtained from the sum of a trend model and signal.

The method of predicting the signal over a surface is a surface-fitting technique (SFT). The ones considered in case studies for this thesis were

- Least-squares collocation.
- Radial basis functions.
- Inverse distance to a power weighting.
- Hybrid inverse power function embodying accelerated decline (HIPFEAD).
- Nearest neighbour interpolation.
- Linear interpolation variant on nearest neighbour (LIVONN).

14.5.1 Not-strictly-bounded SFTs

Of the SFTs listed above, least-squares collocation and radial basis functions are not-strictly-bounded (NSB). This property enables the kind of interpolation illustrated in Figure 7-1 where there are acceptable extremities outside the range of the points being interpolated.

In this study, radial basis functions were applied to a signal component of the datum shift, the signal being the residual shift after a trend model has been removed. This revealed a clear analogy between radial basis functions and "errorless" least-squares collocation. This has been developed with the concept of "revamped signals" (Sections 7.1 and 7.6). Once computed and stored, no further solution of matrix equations is required. Either method can be applied directly rather than through gridding and interpolation.

The covariance functions applied for least-squares collocation (Section 7.1) included two finite functions from Sansò and Schuh (1987) as well the more commonly-used Gaussian function. SS20 has the advantage of simplicity over SS30 but it produces matrices that are less well-conditioned. In fairness to Sansò and Schuh, SS20 was proposed for one-dimensional use rather than surface-fitting.

The above three types of covariance function all require information about correlation at one or more distances. For Western Australia, 82 data points was insufficient to determine this empirically, with the result that experimental methods were used to estimate the second parameter (signal variance being the first parameter). For Slovenia and Great Britain, there were over 3000 data points, and covariance length was estimated from products of signals

(Sections 12.2 and 13.2). The precise process used in this study is not the only sampling method that has been proposed for second-parameter determination but is recommended for its comparative simplicity.

Of the radial basis functions applied, inverse multiquadric gave the best fit at test points for the large datasets in Chapters 12 and 13. The other RBFs, from second-best to worst, were multilog, multiquadric, thin plate spline and natural cubic spline.

One problem with NSB SFTs is that clusters of points which are close together can cause instability. This is illustrated in Figure 7-2.

One possible solution is described in Section 7.13.1. It involves a modified NSB consisting of the sum of

- the NSB obtained after each cluster is temporarily replaced by a pseudo data point, and
- HISFEADs defined around each cluster.

The above function has C^1 continuity and interpolates all the data points without volatility around the clusters. It was used successfully on the Western Australia dataset, where two data points were 89m apart (Section 8.3). Satisfactory datum shifts were obtained for least-squares collocation and 3 of the radial basis functions.

For the Great Britain (2D) dataset, there were points too close to permit this approach. The data made the assumption of no noise untenable. Accordingly, least-squares collocation was successfully applied with a small variance assumed for noise. The finite covariance function SS30 proved better than Gauss (Section 13.2).

For the Slovenia (2D) dataset, clusters were not considered. This was because the shortest distance between data points relative to the longest distance was much larger than for Great Britain (2D) and Western Australia. It is possible that allowance for noise would have produced a better fit for least-squares collocation at the test points, albeit with the loss of an exact fit at the control points. As it was, LSC was only successful when SS30 was the covariance function.

Clusters can be avoided altogether by transferring all but one control point from each cluster to the set of test points. This option is discussed in Section 14.7.

14.5.2 Strictly-bounded SFTs

Strictly-bounded surface-fitting techniques avoid undesirable oscillations because they rely on non-negative weights that add up to 1. That weighted-average property is also a limitation because it prevents even the modest amount of extrapolation shown in Figure 7-1, even when it is justified by the data.

Inverse distance to a power (Section 7.5) has the characteristics illustrated in Figure 7-9. It was considered on this study for each of the powers 1, 2, 3 and 4. The power 1 proved to be poor for interpolation, and in any case the surface generated was not smooth. As the power was increased to 2, 3 and 4, the case studies showed increasingly close fits to the data. However, the improvement was offset by increasingly sharp changes in gradient (evident in Figure 7-9 for n=3 and n=4).

Hybrid inverse power function embodying accelerated decline [HIPFEAD] (Section 7.13) is an original variation on inverse distance to a power. Its subtypes are HISFEAD (subsection 7.13.1) and HICFEAD (subsection 7.13.2). Like finite covariance functions, HIPFEAD removes the influence of control points beyond a certain distance r_{max}. It also accelerates the decline in influence as distance to control points approaches r_{max}. These properties increase the influence of the closest control points.

HISFEAD (subsection 7.13.1) was designed to be an improvement on inverse distance to the power 2 (inverse square distance); in the case studies, it provided a closer fit to the data by between 20% and 52%. While not as accurate as inverse distance to the powers 3 and 4, HISFEAD avoided the sharp changes in gradient associated with those powers.

HICFEAD (subsection 7.13.2) was designed to be an improvement on inverse distance to the power 3 (inverse cubic distance); in the case studies, it provided a closer fit to the data by between 3% and 11%. HICFEAD is also more accurate than HISFEAD, although it does have changes of curvature comparable to inverse cubic distance.

Nearest neighbour (Section 7.7) generates discontinuous signals and would represent datum shifts as a set of flat areas were it not for the effect of the trend model. In this study it was considered purely as a gridding method to be followed by bilinear or bicubic interpolation. In the case studies, bilinear interpolation gave a 4%-9% better fit than bicubic interpolation, despite not being smooth.

LIVONN (Section 7.14) is a new variation on nearest neighbour, and it takes account of the second-nearest control point. For both of the large datasets, LIVONN gives better results than nearest neighbour, whether or not the predicted datum shifts at the test points are obtained via gridding. In the case studies, bilinear interpolation gave a 2%-4% better fit than bicubic interpolation, despite not being smooth.

When gridding methods are used, RMS differences at control points is not a very good indication of accuracy. For the two large datasets, the ratio of RMS differences for test points versus control points was substantially different for Slovenia and Great Britain.

14.5.3 Considerations about gridding

One characteristic of composite methods is that they can be used with or without gridding. The arguments for and against gridding can be summarised as follows.

Those composite methods which do not generate a continuous surface, notably nearest neighbour and (to a lesser extent) LIVONN, must be combined with gridding if a continuous surface is required. This was done in Sections 8.3, 12.2 and 13.2.

Gridded datum shifts can be stored and re-used in a way that avoids having to re-use an SFT. However, in the case of least-squares collocation and radial basis functions, storage of the revamped signal vectors is just as effective in minimising future computation if the direct "trend + signal" approach is used. This kind of storage is not needed for inverse square distance to a power or HIPFEAD because neither of those methods requires the solution of a matrix equation.

Gridding introduces an extra stage of approximation compared to the direct "trend + signal" approach. The "accuracy" of this stage can be quantified by RMS differences between the known shifts at the control points and predicted shifts obtained at those points by bilinear or bicubic interpolation. This was done in Chapter 8 for Western Australia where all data points were control points. This measure of accuracy is only relative because it is based solely on the points from which the SFT was derived.

The largest datasets contained sufficient data points for a significant number to be set aside as test points. This was the case for Slovenia (Chapter 12) and Great Britain (Chapter 13). Only

nearest neighbour and LIVONN were used as gridding methods. For Slovenia the RMS differences at the control points were 32% smaller than the RMS differences at the test points. For Great Britain the RMS differences at the control points were 68%-70% smaller than the RMS differences at the test points. On this evidence, RMS differences at control points from gridding not only give a flattering misrepresentation of the true accuracy of the transformation but give little indication of that accuracy.

14.6 Test-point selection

For the two largest datasets, the data points were divided into control points and test points.

In the case of D48→D96 for Slovenia (Chapter 12), the area was divided into rectangles separated by corridors, and the first data point within each rectangle was selected as a test point (illustrated in Figure 12-2). This ensured that the test points were well spread out, although only 1 in 16 were selected.

In the case of ETRS89→OSGB36 for Great Britain (Chapter 13), the land area was divided into squares separated by corridors as illustrated in Figure 13-3. Up to 3 test points (the 4th, 10th and 20th data points) were taken from each square. This also ensured that the test points were well spread out. Out of the 3965 non-boundary points, 1 in 13 were selected as test points.

The selection method used for Great Britain was the better one, partly because it generated a higher proportion of test points and partly because there was less risk of an isolated data point being omitted from the control points.

14.7 Recommendations for further work

There are a number of ways in which this investigation of datum transformations could be extended. The following are identified in this section:

- Application of datum transformation methods to other datasets.
- Adaptation of split-zone multiple regression equations to areas whose furthest extent is oblique.
- Other radial basis functions, particularly those with possible surface-fitting similarities to least-squares collocation.
- Variations on the test-point selection methods used in the case studies.

The methods studied in this thesis could be applied to datasets other than those used in Chapters 8 to 13. Some datasets, like those described in sub-appendix C.7, can be found in published research papers. These tend to be relatively small. To obtain larger datasets it is necessary to approach contacts in agencies that are not too protective of their data. More extensive testing of the new methods should reveal more information as to which ones are best suited to datasets with particular characteristics.

The methods would also benefit from being tested on simulated datasets. This was done with the data in Appendix D for deriving conformal transformations. It would have been done for the methods of deriving 8-parameter and 9-parameter affine transformations had the evidence from a diverse collection of actual datasets not been so overwhelming. If further study is undertaken to detect gross errors that affect derived transformations, it would be advisable to try to simulate the kinds of data corruption that occur in practice.

The generalisation of multiple regression equations indicated that there were advantages in a zone-split that was a perpendicular bisector of whichever was greater of the N/S extent and the E/W extent. Further research could examine whether this applies to the greatest extent when the area of coverage is oblique, as in Figure 14-1.

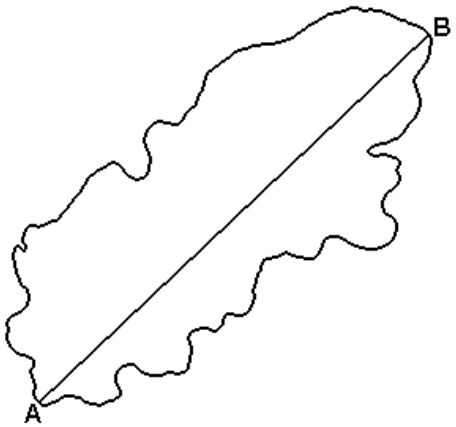


Figure 14-1: Example of an area with a very oblique furthest extent.

The partitioning of such an area would be as shown in Figure 14-2. The *U*-axis would be parallel to the line AB of greatest extent and the *V*-axis would be perpendicular to it. The offset coordinates ϕ_{off} & λ_{off} would be ϕ & λ at that point where $U = V = 0$.

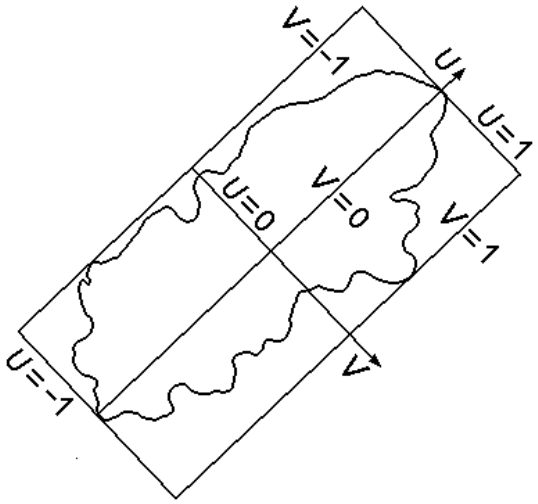


Figure 14-2: Example of intermediate coordinates for MREs created by oblique partitioning.

The equations of the intermediate coefficients would be

$$U = K_1[\phi_{\text{in deg}} - \phi_{\text{off}} + K_3(\lambda_{\text{in deg}} - \lambda_{\text{off}})] \tag{14-1}$$

and

$$V = K_2[\lambda_{\text{in deg}} - \lambda_{\text{off}} - K_3(\phi_{\text{in deg}} - \phi_{\text{off}})], \tag{14-2}$$

where $K_3 = (\lambda_{\text{B}} - \lambda_A)/(\phi_{\text{B}} - \phi_{\text{A}})$, and K_1 & K_2 are normalising constants. The MREs would have the same form as the North/South MREs (subsection 2.17.2) although the zones $V \geq 0$ and $V < 0$ would no longer be north and south.

Composite methods using radial basis functions should be investigated with RBFs other than those listed in Section 7.6. The Gaussian function $\varphi(r) = \exp(-k^2r^2)$ is one such RBF; it is mentioned, for example, in Bullinaria (2015a) and Ziggah *et al* (2018). Like the inverse multiquadric function, it tends to zero as $r \to \infty$.

One particular aspect of the Gaussian RBF worth studying is whether its application is equivalent to the use of errorless least-squares collocation with a Gaussian covariance function. If it is, it would establish an overlap between LSC and RBFs. It would open up a further avenue of investigation: whether there are other RBFs that give equivalent results to LSC when used in covariance functions.

Although the test-point selection method used for Great Britain worked better than the one used for Slovenia, two modifications are recommended for further study.

- The number of points selected could be increased by, for example, selecting the 4th, 9th, 15th and 22nd data points from each square rather than the 4th, 10th and 20th.
- To prevent clusters, *ie* avoid control points that are unacceptably close, each control point could be tested and each unacceptably-near neighbour could be transferred to the test points. Only the first point encountered in each cluster would be retained as a control point. This does, of course, require a decision on what "unacceptably close" means. It is complicated by the fact that moderately close control points only cause a problem if the change in datum shift is large over the distance separating the points.

This thesis also makes two general recommendations about future work on transformations. One is the precise use of terminology that – for example – distinguishes near-conformal models (based on approximations) from conformal models. The other is the clear declaration of adopted conventions coupled with an acknowledgement of the alternatives used elsewhere. This applies particularly to type of rotation parameter (position vector vs coordinate frame), type of Helmert transformation (Version 1 vs Version 2) and type of Bursa-Wolf (fully-linear vs partially-linear). Greater attention to these points – as advocated by Ruffhead and Whiting (2020) – would reduce the confusion caused by much of the existing literature.

REFERENCES

Abbey, D. A., and Featherstone, W. E., 2020. Comparative review of Molodensky-Badekas and Burša-Wolf methods of coordinate transformation. *Journal of Surveying Engineering*, 146 (3), 7 pages.

Agustan and Featherstone, W. E., 2004. Reprocessing the Western Australian Statefix Geodetic GPS Network Using Commercial Software. *Paper presented at the 3rd FIG Regional Conference*, Jakarta, Indonesia, October 3-7, 2004. 14 pages.

Ahmed, A. E. M., 2013. Common Point Coordinates Transformation Parameters Between Adindan (Sudan) New Ellipsoid And The World Geodetic System 1984 (GPS Datum) Coordinates Compared With Parameters Of The American National Imagery And Mapping Agency (NIMA). *International Journal of Advanced Research in Engineering and Applied Sciences*, 2 (9) 26-39.

Aktuğ, B., 2009. Inverse and Compound Datum/Frame Transformations. *Journal of Surveying Engineering*, 135 (2) 46-55.

Al Marzooqi, Y., Fashir, H., and Ahmed, S. I., 2005. Derivation of Datum Transformation Parameters for Dubai Emirate. From Pharaohs to Geoinformatics, FIG Working Week 2005 and GSDI-8, Cairo, Egypt, 16-21 April 2005.

Amiri-Simkooei, A. R., 2018. Parameter estimation in 3D affine and similarity transformation: implementation of variance component estimation. *Journal of Geodesy*, 92 (11), 1285-1297.

Andrei, C. O., 2006. *3D affine coordinate transformations*. MSc thesis. School of Architecture and the Built Environment. Royal Institute of Technology (KTH), Stockholm, Sweden.

Appelbaum, L.T., 1982. Geodetic datum transformation by multiple regression equations. *Proceedings of the Third International Geodetic Symposium on Satellite Doppler Positioning*. Las Cruces, New Mexico, 207-223.

Arabelos, D., and Tscherning, C. C., 1996. CIGAR IV – Workpackage 3: Support of Spaceborne Gravimetry Data Reduction by Ground Based Data. CIGAR IV, Final Report, 95-156.

Arya, S., Mount, D. M., Netenyahu, N. S., Silverman, R., and Wu, A. Y., 1998. An optimal algorithm for approximate nearest neighbour searching fixed dimensions. *Journal of the ACM*, 45 (6), 891-923.

Attaouia, B., Salem, K., Boualem, G., and Bachir, G., 2017. Computation of Continuous Displacement Field from GPS Data – Comparative Study with Several Interpolation Methods. Conference Paper (FIG Working Week 2017: Surveying the world of

tomorrow-From digitalisation to augmented reality May 29 - June 2, Helsinki Finland). 11 pages.

Awange, L. J., and Grafarend, W. E., 2002. Linearized Least Squares and nonlinear Gauss-Jacobi combinatorial algorithm applied to the 7-parameter datum transformation $\mathbb{C}_7(3)$ problem. *Zeitschrift für Vermessungswesen*, 127 (2), 109-116.

Awange, J. L., Bae, K.-H., and Claessens, S, J., 2008. Procrustean solution of the 9-parameter transformation problem. *Earth, Planets and Space*, 60, 529-537.

Ayer, J., and Fosu, C., 2008. Map Coordinate Referencing and the Use of GPS Datasets in Ghana. *Journal of Science and Technology*, 28 (1), 116-127.

Ayer, J., and Tiennah, T., 2008. Datum Transformation by the Iterative Solution of the Abridging Inverse Molodensky Formulae. *The Ghana Surveyor*, 1 (2), 59-66.

Ayer, J., Boakye-Yiadom, E., and Smith-Ephraim, A., 2010. A Java Program for Direct Conversion between the Ghana National Grid and the Universal Transverse Mercator Projection Coordinates.

Baxter, B. J. C., 1992. The interpolation theory of radial basis functions. Doctoral dissertation. Cambridge University.

Berk, S., and Komadina, Ž., 2013. Local to ETRS89 datum transformation for Slovenia: triangle-based transformation using virtual tie points. *Survey Review*, 45 (328), 25-34.

Blais, J. A. R., 1982. Synthesis of Kriging estimation methods. *Manuscripta Geodaetica*, 7 (4), 325-352.

Bohling, G. C., 2005a. *Introduction to geostatistics and variogram analysis*. Kansas Geological Survey, Kansas University. Available at http://people.ku.edu/~gbohling/cpe940.

Bohling, G. C., 2005b. *Kriging*. C&PE 940, Kansas Geological Survey, Kansas University. 20 pages.

Bomford, G., 1980. *Geodesy* (fourth edition). Oxford, Clarendon Press.

Borkowski, K. M., 1989. Accurate algorithms to transform geocentric to geodetic coordinates. *Bulletin Géodésique*, 63(1), 50-56.

Bowring, B. R., 1976. Transformation from Spatial to Geographical Coordinates. *Survey Review*, 23 (181), 323-327.

Briggs, I.C., 1974. Machine contouring using minimum curvature. *Geophysics*, 39 (1), 39-48.

Brunei Darussalam Survey Department, 2009. *A Technical Manual of the Geocentric Datum Brunei Darussalam 2009 (GDBD2009)*. Version 1.1. 61 pages.

Bullinaria, J. A., 2015a. Radial Basis Function Networks: Introduction. Neural Computation: Lecture 13. 16 pages. https://www.cs.bham.ac.uk/~jxb/INC/l13.pdf

Bullinaria, J. A., 2015b. Radial Basis Function Networks: Algorithms. Neural Computation: Lecture 14. 18 pages. https://www.cs.bham.ac.uk/~jxb/INC/l14.pdf

Bursa, M., 1962. The Theory of the Determination of the Non-parallelism of the Minor Axis of the Reference Ellipsoid and the Inertial Polar Axis of the Earth, and the planes of the Initial Astronomic and Geodetic Meridians from Observation of Artificial Earth Satellites. *Studia Geophysica et Geodetica*, 6 (3), 209-214.

Chen, W., and Hill, C., 2005. Evaluation Procedure for Coordinate Transformation. *Journal of Surveying Engineering*, 131 (2), 43-49.

Cross, P. A., Hollway, J. R., and Small, L. G., 1982. *Geodetic Appreciation*. Working Paper No. 2, Department of Land Surveying, North East London Polytechnic. 198 pages.

Cross, P. A., 1983. *Advanced Least Squares Applied to Position-fixing.* Working Paper No. 6, Department of Land Surveying, North East London Polytechnic. 173 pages.

Dawod, G.M., and Alnaggar, D.S., 2000. Optimum Geodetic Datum Transformation Techniques for GPS Surveys in Egypt. *Proceedings of Al-Azhar Engineering Sixth International Engineering Conference*, Al-Azhar University, 1-4 September, 4, 709-718.

Dawson, J., and Woods, A., 2010. ITRF to GDA coordinate transformation. *Journal of Applied Geodesy*, 4 (4), 189-199.

Deakin, R. E., Collier, P. A., and Leahy, F. J., 1994. Transformation of Coordinates using Least Squares Collocation. *Australian Surveyor,* 39 (1), 6-20.

Deakin, R. E., 2004. *The Standard and Abridged Molodensky Coordinate Transformation Formulae*. Department of Mathematical and Geospatial Sciences, RMIT University, Melbourne, Australia, 22 pages.

Deakin, R. E., 2006. *A Note on the Bursa-Wolf and Molodensky-Badekas Transformations*. Department of Mathematical and Geospatial Sciences, RMIT University, Melbourne, Australia, 21 pages.

Dermanis, A., 1984. Kriging and collocation: a comparison. *Manuscripta Geodaetica*, 9 (3), 159-167.

Dewhurst, W.T., 1990. The application of minimum curvature-derived surfaces in the transform of positional data from the North American Datum of 1927 to the North American Datum of 1983. NOAA Technical Memorandum NOS NGS-50.

Dewitt, B., 1996. Initial approximations for the three-dimensional conformal coordinate transformation. *Photogrammetric Engineering and Remote Sensing*, 62 (1), 79–83.

Dhungana, R., and Lama, S., 2018. *Final Project Report on The determination of transformation parameter*. Government of Nepal Ministry of Land Reform and Management, 44 pages.

DMA, 1987a. *Supplement to Department of Defense World Geodetic System 1984 Technical Report: Part I – Methods, Techniques, and Data Used in WGS 84 Development*, DMA TR 8350.2-A, National Imagery and Mapping Agency, Washington, WA, USA.

DMA, 1987b. *Supplement to Department of Defense World Geodetic System 1984 Technical Report: Part II – Parameters, Formulas, and Graphics for the Practical Application of WGS 84*, DMA TR 8350.2-B, National Imagery and Mapping Agency, Washington, WA, USA.

DMA, 1991. *Department of Defense World Geodetic System 1984: Its definition and relationships with local geodetic systems.* (Second Edition), Technical Report no. 8350.2, Defense Mapping Agency.

Doytsher, Y., 2000. A rubber sheeting algorithm for non-rectangular maps. *Computers & Geosciences*, 26 (9-10), 1001-1010.

Dzidefo, A., 2011. *Determination of transformation parameters between the World Geodetic System 1984 and the Ghana Geodetic Network*. Master's Thesis, Department of Civil and Geomatic Engineering, Kwame University of Science and Technology, Kumasi, Ghana. 97 pages.

El-Mewafi, M., 2015. *A New Transformation Parameters Algorithm for Egypt using Finite Elements*. 14 pages.

Engberg, L. E., and Lilje, M., 2002. A New Co-ordinate System for Sweden. TS5.5 *Reference Frame in Practice*, FIG XXII International Congress Washington, D.C. USA, April 19-26 2002.

ESRI, 2004. Supported geographic coordinate systems. downloads.esri.com/support/whitepapers/ao_/geographic_coordinate_systems9.pdf

ESRI, 2008. ArcGIS Desktop 9.3 Help: Equation-based methods. [http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=Equation-based%20methods]

ESRI, 2012. ArcGIS 10.1 Geographic and Vertical Transformation Tables. resources.arcgis.com/en/help/main/10.1/003r/pdf/geographic_transformations.pdf

European Union, 2013. Methodology and Parameters for Datum Transformation between the New and Old Reference Systems (The European Union's ENPI Programme for Georgia). 29 pages.

Fan, J., and Gijbels, I., 1996. *Local polynomial modelling and its applications: Monographs on statistics and applied probability*, Volume 66, Chapman and Hall, London.

Fan, H., 2005. Three-Dimensional Coordinate Transformations with large rotations and scale change. International Workshop on Education in Geospatial Information Technology, 27-28 October, Technical University of Moldova, Chisinau, Moldova.

Fan, H., 2009. Direct solution of three-dimensional coordinate transformation parameters. Department of Geodesy, Royal Institute of Technology (KTH), Sweden.

Fang, X. 2014. A total least squares solution for geodetic datum transformations. *Acta Geodaetica et Geophysica*, 49 (2), 189-207.

Featherstone, W., 1997. A comparison of existing coordinate transformation models and parameters in Australia. *Cartography*. 26 (1), 13-26.

Featherstone, W. E., Galloway, D., Goulding, P., and Reit, B.-G., 1999. Transformation between Australian datums using a modified transverse Mercator projection. *Cartography*, 28 (1), 19-32.

Featherstone, W. E., and Claessens, S. J., 2008. Closed-Form Transformation Between Geodetic and Ellipsoidal Coordinates. *Studia Geophysica et Geodaetica*, 52 (1), 1-18.

Forsberg, R., and Tscherning, C. C. T., 2008. *An overview manual for the GRAVSOFT geodetic gravity field modelling programs*, DTU Space, Kongens Lyngby, Denmark.

Franke, R., and Nielson, G., 1980. Smooth interpolation of large sets of scattered data. *International Journal for Numerical Methods in Engineering*, 15, 1691-1704.

Fukushima, T., 1999. Fast transform from geocentric to geodetic coordinates. *Journal of Geodesy*, 73, 603-610.

Gacoki, T. G., and Aduol, F. W. O., 2002. Transformation between GPS coordinates and local plane UTM coordinates using the Excel spreadsheet. *Survey Review*, 36 (284), 449-462.

Ge, Y., Yuan, Y., and Jia, N., 2013. More efficient methods among commonly used robust estimation methods for GPS coordinate transformation. *Survey Review*, 45 (330), 229-234.

Gillman, W.D., 1985. Triangulations for rubber sheeting. *Proceedings of the 7th International Symposium on Computer Assisted Cartography.* American Society for Photogrammetry and Remote Sensing and the American Congress on Surveying and Mapping, Washington, DC. 191–199.

Gledan, J.A., and Azzeidani, O.A., 2014. ELD79-LGD2006 Transformation Techniques – Implementation and Accuracy Comparison in Tripoli Area, Libya. *International Journal of Civil, Environmental, Structural, Construction and Architectural Engineering*, 8, 255-258.

González-Matesanz, J., Dalda, A., Quirós, R., and Celada, J., 2003. ED50-ETRS89 Transition Models for the Spanish Geodetic Network. Paper presented at 2003 EUREF Symposium.

González-Matesanz, J., Dalda, A., and Malpica, J. A., 2006. A range of ED50-ETRS89 datum transformation models tested on the Spanish Geodetic Network. *Survey Review* 38 (302), 654-667.

Gower, J. C., and Dijksterhuis, G. B., 2004. *Procrustes Problems*, New York, Oxford University Press. 248 pages.

Greaves, M., 2004. OSTN02: A new definition transformation from GPS derived coordinates to national grid coordinates in Great Britain. *Survey Review*, 37 (293), 502-519.

Greaves, M., Downie, P., and Fitzpatrick, K., 2016. OSGM15 and OSTN15: Updated transformations for UK and Ireland. Geomatics World, July/August 2016.

Grgić, M., Varga, M., and Bašić, T., 2016. Empirical research of interpolation methods in distortion modeling for the coordinate transformation between local and global geodetic datums. *Journal of Surveying Engineering* 142 (2), article number 05015004, 9 pages.

Han, J.-Y., 2010. Noniterative approach for solving the indirect problems of linear reference frame transformations. *Journal of Surveying Engineering*, 136 (4), 150-156.

Hardy, R. L., 1990. Theory and Applications of the Multiquadric-BiHarmonic Method – 20 Years of Discovery 1968-1988. *Computers & Mathematics with Applications*, 19 (8/9) 163-208.

Harvey, B. R., 1986. Transformation of 3D Coordinates. *The Australian Surveyor*, 33 (2), 105-125.

Hashemi, A., Kalantari, M., and Kasser, M., 2013. Direct solution of the 7 parameters transformation problem. *Applied Mathematics & Information Sciences*, 7 (4), 1375-1382.

Heiskanen, W.A., and Moritz, H., 1967. *Physical Geodesy*. W. H. Freeman San Francisco.

Hofmann-Wellenhof, B., and Moritz, H., 2006. *Physical Geodesy (Second Edition)*. SpringerWienNewYork.

ICSM, 2014. *Geocentric Datum of Australia Technical Manual* Version 2.4. 62 pages.

Iliffe, J. C., and Lott, R., 2008. *Datums and map projections: for remote sensing, GIS and surveying* (2nd edition) Whittles Publishing.

Ipbüker, C., and Bildirici, I. Ö., 2002. A general algorithm for the inverse transformation of map projections using Jacobian matrices. *Proceedings of the Third International Symposium Mathematical & Computational Applications September 4-6, 2002*. Konya, Turkey, 175-182.

Ipbüker, C., 2009. Inverse Transformation for Several Pseudo-cylindrical Map Projections Using Jacobian Matrix. Conference paper. ICCSA 2009, Part 1, LNCS 5592, 553-564.

ISO, 2007. *ISO 19111, Geographical information – Spatial referencing by coordinates*, Second Edition, International Organization for Standardization.

Jonsson, B., Hedling, G., and Wiklund, P., 2003. *SWEPOS™Network-RTK Services – status, applications and experiences.* Presented at ION GPS/GNSS 2003, 9-12 September, 2003, Portland, Oregon, US.

Junkins, D.R., and Farley, S.A., 1995. *NTv2 User's Guide*, Geodetic Survey Division, Natural Resources Canada.

Kashani, I., 2006. Application of Generalized approach to Datum Transformation between Local Classical and Satellite-based Geodetic Networks. *Survey Review*, 38 (299), 412-422.

Keckler, D., 1995. *The Surfer Manual*. Golden Software, Inc.: Golden, Colorado, USA.

Kinneen, R.W., and Featherstone, W.E., 2004. Empirical Evaluation of Published Transformation Parameters from the Australian Geodetic Datums (AGD66 and AGD84) to the Geocentric Datum of Australia (GDA94). *Journal of Spatial Science* 49 (2), 1-29.

Knippers, R. A., 2009. Coordinate Transformations: Chapter 5 from *Geometric Aspects of Mapping*, International Institute for Geo-Information Science and Earth Observation (ITC), Enschede.
http://kartoweb.itc.nl/geometrics/Coordinate%20transformations/coordtrans.html

Kumi-Boateng, B., and Ziggah, Y. Y., 2020. Feasibility Of Using Group Method Of Data Handling (GMDH) Approach For Horizontal Coordinate Transformation. *Geodesy and Cartography*, 46 (2), 55-66.

Kutoglu, H. S., 2009a. Direct determination of local coordinates from GPS without transformation. *Survey Review*, 41 (312), 162-173.

Kutoglu, H.S., 2009b. Alternative methods for improving Transformation Consistency between Geocentric and Non-Geocentric Coordinate Systems. *Survey Review*, 41 (314), 408-418.

Laari, P. B., Ziggah, Y. Y., and Annan, R. F., 2016. Determination of 3D Transformation Parameters for the Ghana Geodetic Reference Network using Ordinary Least Squares and Total Least Squares Techniques. *International Journal of Geomatics and Geosciences*, 7 (3), 245-261.

Lancaster, P., and Salkauskas, K., 1981. Surfaces generated by moving least squares methods. *Mathematics of Computation*, 37 (155), 141-158.

Lazzaro, D., and Montefusco, L. B., 2002. Radial basis functions for the multivariate interpolation of large scattered data sets. *Journal of Computational and Applied Mathematics*, 140, 521-536.

Lee, D. T., and Schachter, B. J., 1980. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer and Information Science*, 9 (3), 219-242.

Leick, A., 1995. *GPS Satellite Surveying, Second Edition.* John Wiley & Sons, Inc. 580 pages.

Malys, S., 1988. *Dispersion and Correlations among Transformation Parameters, Relating Two Satellite Reference Frames*. Report No. 329. Dept. of Geodetic Science and Surveying, The Ohio State University, Columbus, Ohio. 95 pages.

McCarthy, D. D. (ed), 1996. IERS Conventions (1992). Observatoire de Paris, IERS Technical Note 21.

Merry, C.L., and Whittal, J.F., 1998. Transformation of Coordinate Data for the New South African Datum – A Pilot Study. *Survey Review* 34 (268), 379-388.

Menz, J., Benndorf, J., and Wambeke, T., 2015. Signal-Controlled Least Square Collocation: A new quality in geostatistical estimation and simulation? *Spatial Statistics*, 14 (C), 303-320.

Mitsakaki, C., Agatza-Balodimou, A.M., and Papazissi, K., 2006. Geodetic Reference Frames Transformations. *Survey Review*, 38 (301), 608-618.

Mohammed, A. E., and Mohammed, N. Z., 2013. WGS84 to Adindan-Sudan Datum Transformation Manipulated by ITRF96. *International Journal Of Multidisciplinary Sciences And Engineering*, 4 (5), 60-64. [NB: 1st author is also known as A. E. M. Ahmed; 2nd author is also known as Nagi Zomrawi.]

Molnár, G., and Timár, G., 2005. Determination of the parameters of the abridging Molodensky formulae providing the best horizontal fit. *Geophysical Research Abstracts*, 7 (01018).

Molodensky, M. S., Eremeev, V. F., and Yurkina, M. I., 1962. *Methods for Study of the External Gravitational Field and Figure of the Earth*. Translated from Russian. Israel Programme for Scientific Translations.

Musashi, J. P., Pramoedyo, H., and Fitrian, R., 2018. Comparison of Inverse Distance Weighted and Natural Neighbor Interpolation Method at Air Temperature Data in Malang Region. *Cauchy – Jurnal Matematika Murni Dan Aplikasi*, 5 (2), 48-54.

NATO, 2001. Standardization Agreement (STANAG) 2211 IGEO, *Geodetic Datums, Projections, Grids and Grid References*, Edition 6, NATO Military Agency for Standardization, Brussels.

NGS, 2013. *NADCON User Manual*. US Department of Commerce National Geodetic Service.

NGA, 2008. NATO datum transformation parameters. (Available as http://earth-info.nga.mil.GandG/coordsys/datum/NATO_DT.pdf)

NGA, 2014. *Department of Defense World Geodetic System 1984: its definition and relationships with local geodetic systems*. NGA-STND.0036_1.0.0_WGS84, Version 1.0.0.

NIMA, 2000. *Department of Defense World Geodetic System 1984: Its definition and relationships with local geodetic systems.* (Third Edition), Technical Report no. 8350.2. Washington: National Imagery and Mapping Agency. [NB the date is that on the cover, although it contains Amendment 2 dated 23 June 2004.] http://gis-lab.info/docs/nima-tr8350.2-wgs84fin.pdf

Okwuashi, O., and Eyoh, A. 2012. 3D coordinate transformation using total least squares. *Academic Research International*, 3 (1), 399-405.

Ordnance Survey, 2002. *Transformations and OSGM02 User guide*. Version 2.1. Ordnance Survey, UK. 43 pages.

Ordnance Survey, 2018. *A guide to coordinate systems in Great Britain*. Version 3.3. https://www.ordnancesurvey.co.uk/docs/support/guide-coordinate-systems-great-britain.pdf?awc=2495_1472758581_2be7907c343c32b09a8d5171103197d7

Ordnance Survey, 2020. *Transformations and OSGM15 User guide*. Version 1.3. Ordnance Survey, UK. 30 pages.

Paláncz, B, Lewis, R. H., Zaletnyik, P., and Awange, J., 2008. *Computational Study of 3D Affine Coordinate Transformation. Part 1. 3-point Problem*. 25 pages. [https://home.bway.net/lewis/affine1.pdf]

Paláncz, B., Zaletnyik, P., Awange, J. L., and Heck, B., 2010. Extension of the ABC-Procrustes algorithm for 3D affine coordinate transformation. *Earth, Planets and Space*, 62 (11), 857-862.

Paláncz, B and Piroska, Z., 2011. A symbolic solution of a 3D affine transformation. *The Mathematica Journal*, 13, 1-15.

Paláncz, B., Awange, J., and Völgyesi, L., 2017. A novel RANSAC approach to robustly solve the 3D similarity problem. *Australian Journal of Earth Sciences*, 64 (4), 565-576.

Pan, G., Zhou, Y., Sun, H., and Guo, W., 2015. Linear observation based least squares. *Survey Review*, 47 (340), 18-27.

Paul, M. K., 1973. A note on computation of geodetic coordinates from geocentric (Cartesian) coordinates. *Bulletin Géodésique*, 108 (1), 135-139.

Powell, M. J. D., 2005. Five Lectures on Radial Basis Functions. TeXified version of hand-outs (made by Hans Bruun Nielsen, IMM) for five lectures given at IMM in 2004. 27 pages. www2.imm.dtu.dk/pubdb/views/edoc_download.php/3600/pdf/imm3600.pdf

Rabah, M., Elmewafey, M., and Farahan, M. H., 2016. Datum maintenance of the main Egyptian geodetic control networks by utilizing Precise Point Positioning "PPP" technique. *NRIAG Journal of Astronomy and Geophysics*, 5 (1), 96-105.

Redfearn, J. C. B., 1948. Transverse Mercator Formulae. *Empire Survey Review*, IX, 318-322.

Rapp, R. H., 1993. *Geometric Geodesy Part II*. Dept. of Geodetic Science and Surveying, The Ohio State University, Columbus, Ohio. 225 pages.

Reit, B.-G., 1998. The 7-parameter transformation to a horizontal geodetic datum. *Survey Review*, 34 (268), 400-404.

Renka, R. J., 1988. Multivariate interpolation of large sets of scattered data. *ACM Transaction on Mathematical Software* 14 (2), 139-148.

Ruffhead, A. C., 1987. An introduction to least-squares collocation. *Survey Review* 29 (224), 85-94.

Ruffhead, A. C., 1998. Enhancement of inverse projection algorithms with particular reference to the Syrian Stereographic projection. *Survey Review*, 34 (270), 501-508.

Ruffhead, A. C., 2016. The SMITSWAM method of datum transformations consisting of Standard Molodensky in two stages with applied misclosures. *Survey Review*, 48 (350), 376-384.

Ruffhead, A. C., 2018. Introduction to multiple regression equations in datum transformations and their reversibility. *Survey Review*, 50 (358), 82-90.

Ruffhead, A. C., and Whiting, B. M., 2020. *Introduction to geodetic datum transformations and their reversibility*. Survey Working Paper No 01/2020, School of Architecture, Computing and Engineering, University of East London. 30 pages.

Ruffhead, A. C., 2021a. Derivation of rigorously-conformal 7-parameter 3D geodetic datum transformations. *Survey Review*, 53 (376), 8-15.

Ruffhead, A. C., 2021b. Equivalence properties of 3D conformal transformations and their application to reverse transformations. *Survey Review*, 53 (377), 158-168. [NB: the source can also be identified as Journal of Geodesy, 61 (4), 331-347.]

Schut, G. H., 1973. Similarity transformation and least squares. *Photogrammetric Engineering*, 39 (6), 621-627.

Shepard, D., 1968. A two-dimensional interpolation function for irregularly-spaced data. *Proceedings of the 1968 ACM National Conference*, Association for Computing Machinery, New York, 517–524.

Sibson, R., 1981. A brief description of natural neighbour interpolation. *Interpolating multivariate data*, ed. V. Barrett, John Wiley, Chichester, U.K., 21-36.

Sjöberg, L. E., 2013. Closed-form and iterative weighted least squares solutions of Helmert transformation parameters. *Journal of Geodetic Science* 3 (1), 7-11.

Snyder, J.D., 1987. *Map Projections — A Working Manual*. U.S. Geological Survey Professional Paper 1395. United States Government Printing Office, Washington. Available online from https://pubs.usgs.gov/pp/1395/report.pdf.

Soler, T., 1976. *On Differential Transformations Between Cartesian And Curvilinear (Geodetic) Coordinates*. Department of Geodetic Science Report No. 236, Ohio State University. 89 pages.

Soler, T., and Hothem, L. D., 1988. Coordinate Systems Used in Geodesy: Basic Definitions and Concepts. *Journal of Surveying Engineering*, 114 (2), 84-97.

Soycan, M., 2005. Polynomial Versus Similarity Transformations Between GPS And Turkish Reference Systems. *Survey Review*, 38 (295), 58-69.

Soycan, M., and Soycan, A., 2008. Transformation of 3D GPS Cartesian Coordinates to ED50 using polynomial fitting by robust re-weighting technique. *Survey Review*, 40 (308), 142-155.

Späth, H., 2004. A numerical method for determining the spatial Helmert transformation in the case of different scale factors. *Zeitschrift für Vermessungswesen*, 129 (4), 255–259.

Stewart, M. P., Houghton, H., and Ding, X., 1998. The Statefix West Australian GPS Network. *Advances in Position and Reference Frames* (ed F. K. Brunner). International Association of Geodesy Symposia, 118, 155-160.

Su, P., and Drysdale, R. L. S., 1997. A comparison of sequential Delaunay triangulation algorithms. *Computational Geometry*, 7 (5-6), 361-385.

Surfer, 2002. Surfer User's Guide: Contouring and 3D Surface Mapping for Scientists and Engineers. Golden, Colorado, USA. 664 pages.

Syetiawan, A., Ramdani, D., Safii, A. N., Ardhitasari, Y., Lumban-Gaol, and Annuriah, I., 2019. Development of Parameter Transformation of Indonesian Geospatial Reference System (2013). Conference Paper. *GEODETA 2019 The 1st International Conference on Geodesy, Geomatics, and Land Administration 2019*. 77-94.

Thomas, T. L., 1947. *Survey in Syria/The Syrian Stereographic Grid*. Royal School of Mines report, Department of Mines, Imperial College of Science and Technology, London, UK. 7 pages.

Timár, G., Baiocchi, V., and Lelo, K., 2011. Geodetic Datums Of The Italian Cadastral Systems. *Geographia Technica*, 06 (1), 82-90.

Tomczak, M., 1998. Spatial Interpolation and its Uncertainty Using Automated Anisotropic Inverse Distance Weighting (IDW) – Cross-Validation/Jackknife Approach. *Journal of Geographic Information and Decision Analysis*, 2 (2), 18-30.

Varga, M., Grgić, M., and Bašić, T., 2017. Empirical comparison of the geodetic coordinate transformation models: a case study of Croatia. *Survey Review*, 49 (352), 15-27.

Veis, G., 1960. Geodetic use of artificial satellites. *Smithsonian Contributions to Astrophysics*, 3 (9), 95-161.

Vermeille, H., 2002. Direct transformation from geocentric coordinates to geodetic coordinates. *Journal of Geodesy*, 76 (8), 451-454.

Vermeille, H., 2004. Computing geodetic coordinates from geocentric coordinates. *Journal of Geodesy*, 78 (1-2), 94-95.

Vermeille, H., 2011. An analytical method to transform geocentric into geodetic coordinates. *Journal of Geodesy*, 85 (2), 105-117.

Wang, Q., Chang, G., Xu, T., and Zou, Y., 2018. Representation of the rotation parameter estimation errors in the Helmert transformation model. *Survey Review*, 50 (358), 69-81.

Watson, G. A., 2005. Computing Helmert transformations. *Journal of Computational and Applied Mathematics*, 197 (2), 387-394.

Wolf, H., 1963. Geometric connection and re-orientation of three-dimensional triangulation nets. Bulletin Géodésique, 68, 165-169.

Wolf, P. R., and Ghalani, C. D., 1997. *Adjustment Computations: Statistics and Least-Squares in Surveying and GIS*. 3rd Edition, John Wiley & Sons, 347-353.

Woodson, J., 2016. How can I remove the bullseye effect that is created in my Surfer grid? Golden Software. https://support.goldensoftware.com/hc/en-us/articles/226504827-How-can-I-remove-the-bullseye-effect-that-is-created-in-my-Surfer-grid-

You, R-.J., and Hwang, H-.W., 2006. Coordinate Transformation between Two Geodetic Datums of Taiwan by Least-Squares Collocation. *Journal of Surveying Engineering*, 132 (2), 64-70.

Yun, H.S., Song, D.S., and Cho, J.M., 2006. Horizontal datum transformation by distortion modelling in Korea. *Survey Review*, 38 (301), 554-562.

Yusheng, G., 2014. Simulation design of Bursa-Wolf coordinate transformation model based on the access. *Computer Modelling & New Technologies*, 18 (2), 83-86.

Závoti, J., and Kalmár, J., 2016. A comparison of different solutions of the Bursa–Wolf model and of the 3D, 7-parameter datum transformation. *Acta Geodaetica et Geophysica*, 51 (2), 245-256.

Zgonc, A., 2006. Application of geodetic datums in georeferencing. *EUMETNET/OPERA 1996-2006*, WD 2005/18. Revision 1.3. 28 pages.

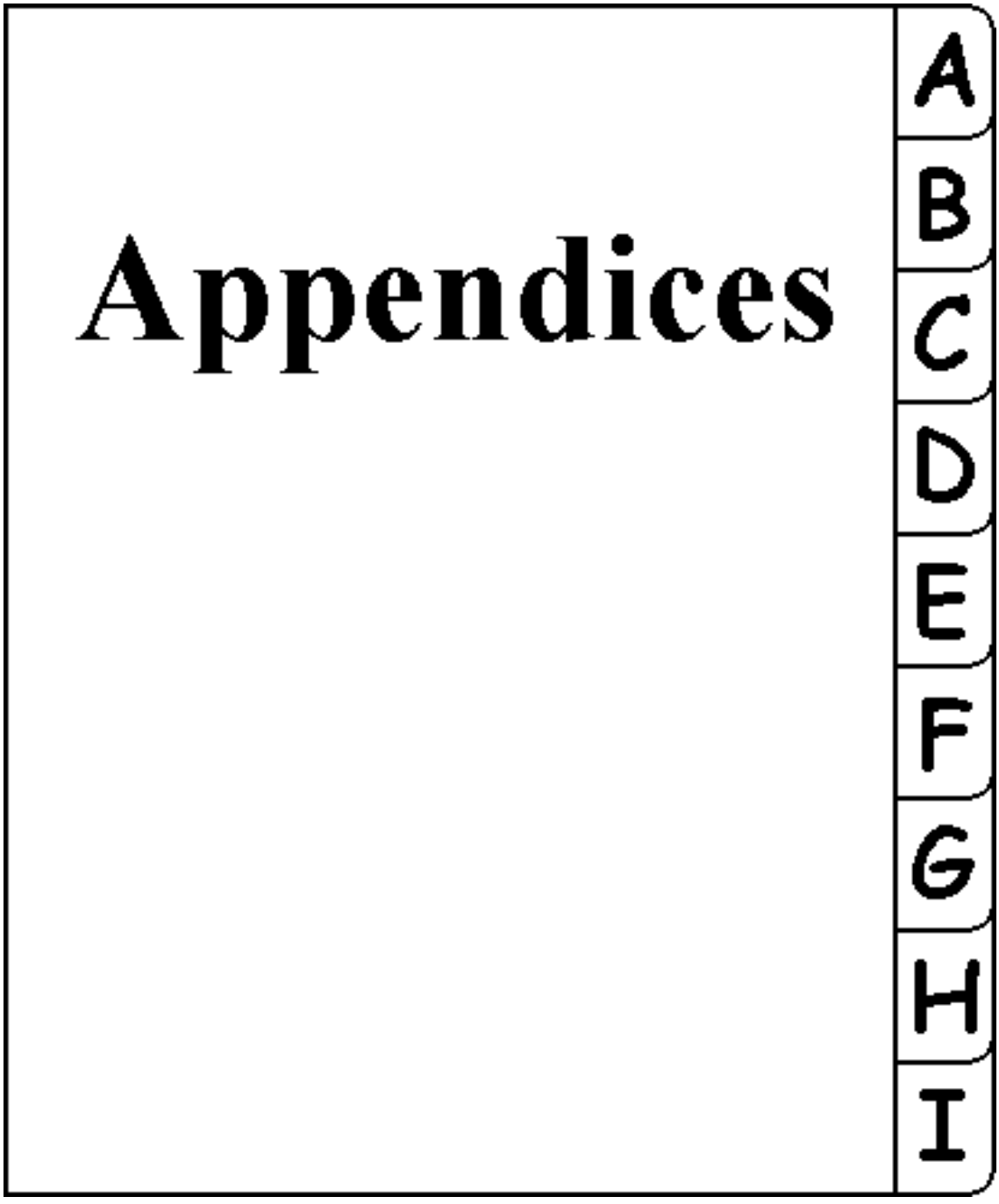
Ziggah, Y. Y., Youjian, H., Amans, C. O., and Fan, D. L., 2013. Determination of GPS Coordinate Transformation Parameters of Geodetic Data between Reference Datums – A Case Study of Ghana Geodetic Reference Network. *International Journal of Engineering Sciences & Research Technology*, 2 (4), 956-971.

Ziggah, Y. Y., Yakubu, I., and Kumi-Boateng, B., 2016. Analysis of Methods for Ellipsoidal Height Estimation – The Case of a Local Geodetic Reference Network. *Ghana Mining Journal*, 16 (2), 1-8.

Ziggah, Y. Y., Hu, Y., Laari, P. B., and Hui, Z., 2017. Novel approach to improve geocentric translation model performance using artificial neural network technology. *Boletim de Ciencias Geodesicas*, 23 (1), 213-233.

Ziggah, Y. Y., Issaka, Y., Laari, P. B., and Hui, Z., 2018. 2D Cadastral Coordinate Transformation using extreme learning machine technique. *Geodesy and Cartography*, 67 (2), 321-343.

Appendices



APPENDIX A: VERIFICATION OF HELMERT EQUIVALENCES

In Section 2.6, there are two versions of the process by which the rigorous 3D rotation matrix in equation (2-14) is computed. Version 1 applies R_X to the source coordinates first and R_Z last, as in equations (2-15) and (2-16). Version 2 applies R_Z to the source coordinates first and R_X last, as in equations (2-17) and (2-18).

A.1 Rotation-parameter conversion from Version 1 to Version 2

Section 2.6.1 describes how to convert Version-1 parameters R_X, R_Y and R_Z into Version-2 parameters R'_X, R'_Y and R'_Z which lead to the same rotation matrix. In so doing, it left the following identities unverified.

$$s'_X s'_Y c'_Z + c'_X s'_Z = r_{2,1}. \tag{A-1}$$
$$c'_X c'_Z - s'_X s'_Y s'_Z = r_{2,2}. \tag{A-2}$$
$$s'_X s'_Z - c'_X s'_Y c'_Z = r_{3,1}. \tag{A-3}$$
$$c'_X s'_Y s'_Z + s'_X c'_Z = r_{3,2}. \tag{A-4}$$

The proofs make use of all the equations in subsection 2.6.1 except for (2-20) because the above identities have still to be proved. They also use the trigonometrical identity $\sin^2\theta + \cos^2\theta = 1$ and its variations, notably $\cos^2\theta - \sin^2\theta = 1 - 2\sin^2\theta$ and $\sin^2\theta - \cos^2\theta = 1 - 2\cos^2\theta$.

The proofs all involve $(c'_Y)^2$ as a common factor in both parts of a quotient. This means it is assumed that c'_Y is non-zero. The special case where c'_Y is zero is covered separately in subsection 2.6.1.

From (2-22),

$$(c'_Y)^2 = 1 - s_X^2 s_Z^2 - c_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z. \tag{A-5}$$

Since $1 - s_X^2 s_Z^2 = (s_Z^2 + c_Z^2) - (1 - c_X^2)s_Z^2$, equation (A-5) can be rewritten as

$$(c'_Y)^2 = c_Z^2 + c_X^2 s_Z^2 - c_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z. \tag{A-6}$$

Proof of (A-1):

$$\begin{aligned}
s'_X s'_Y c'_Z + c'_X s'_Z &= (-r_{2,3}/c'_Y) r_{1,3}(r_{1,1}/c'_Y) + (r_{3,3}/c'_Y)(-r_{1,2}/c'_Y) \\
&= [(s_X c_Z - c_X s_Y s_Z)(s_X s_Z + c_X s_Y c_Z) c_Y c_Z + c_X c_Y (c_X s_Z - s_X s_Y c_Z)]/(c'_Y)^2 \\
&= [(s_X^2 s_Z c_Z^2 + s_X c_X s_Y c_Z^3 - s_X c_X s_Y s_Z^2 c_Z - c_X^2 s_Y^2 s_Z c_Z^2 + c_X^2 s_Z - s_X c_X s_Y c_Z) c_Y]/(c'_Y)^2 \\
&= [(c_X^2 s_Z + s_X^2 s_Z c_Z^2 - c_X^2 s_Y^2 s_Z c_Z^2 + (s_X c_X s_Y c_Z^3 - s_X c_X s_Y c_Z) - s_X c_X s_Y s_Z^2 c_Z) c_Y]/(c'_Y)^2 \\
&\qquad \text{by rearrangement of terms} \\
&= [(c_X^2 s_Z + s_X^2 s_Z c_Z^2 - c_X^2 s_Y^2 s_Z c_Z^2 - s_X c_X s_Y s_Z^2 c_Z - s_X c_X s_Y s_Z^2 c_Z) c_Y]/(c'_Y)^2 \quad \text{as } c_Z^2 - 1 = -s_Z^2 \\
&= [(c_X^2 + s_X^2 c_Z^2 - c_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z) s_Z c_Y]/(c'_Y)^2 \\
&= c_Y s_Z \quad \text{by (A1.6)} \\
&= r_{2,1}.
\end{aligned}$$

Proof of (A-2):

$$\begin{aligned}
c'_X c'_Z - s'_X s'_Y s'_Z &= (r_{3,3}/c'_Y)(r_{1,1}/c'_Y) - (-r_{2,3}/c'_Y) r_{1,3}(-r_{1,2}/c'_Y) \\
&= [(c_X c_Y)(c_Y c_Z) - (s_X c_Z - c_X s_Y s_Z)(s_X s_Z + c_X s_Y c_Z)(c_X s_Z - s_X s_Y c_Z)]/(c'_Y)^2 \\
&= [c_X c_Y^2 c_Z + (c_X s_Y s_Z - s_X c_Z)(s_X c_X s_Z^2 - s_X^2 s_Y s_Z c_Z + c_X^2 s_Y s_Z c_Z - s_X c_X s_Y^2 c_Z^2)]/(c'_Y)^2 \\
&= [c_X c_Y^2 c_Z + s_X c_X^2 s_Y s_Z^3 - s_X^2 c_X s_Y^2 s_Z^2 c_Z + c_X^3 s_Y^2 s_Z^2 c_Z - s_X c_X^2 s_Y^3 s_Z c_Z^2 \\
&\qquad - s_X^2 c_X s_Z^2 c_Z + s_X^3 s_Y s_Z c_Z^2 - s_X c_X^2 s_Y s_Z c_Z^2 + s_X^2 c_X s_Y^2 c_Z^3]/(c'_Y)^2 \\
&= [c_X c_Y^2 c_Z + c_X^3 s_Y^2 s_Z^2 c_Z - s_X^2 c_X s_Z^2 c_Z + s_X^2 c_X s_Y^2 c_Z^3 - s_X c_X^2 s_Y s_Z c_Z^2 \\
&\qquad + s_X c_X^2 s_Y s_Z^3 + s_X^3 s_Y s_Z c_Z^2 - s_X c_X^2 s_Y^3 s_Z c_Z^2 - s_X^2 c_X s_Y^2 s_Z^2 c_Z]/(c'_Y)^2 \\
&\qquad \text{by rearrangement of terms} \\
&= [c_X c_Y^2 c_Z + s_X^2 c_X s_Y^2 s_Z^2 c_Z + c_X^3 s_Y^2 s_Z^2 c_Z + s_X^2 c_X s_Y^2 c_Z^3 - s_X^2 c_X s_Z^2 c_Z - 2 s_X c_X^2 s_Y s_Z c_Z^2 \\
&\qquad + s_X c_X^2 s_Y s_Z^3 + s_X c_X^2 s_Y s_Z c_Z^2 + s_X^3 s_Y s_Z c_Z^2 - s_X c_X^2 s_Y^3 s_Z c_Z^2 - 2 s_X^2 c_X s_Y^2 s_Z^2 c_Z]/(c'_Y)^2 \\
&\qquad \text{by introducing terms that cancel} \\
&= (c_Y^2 + s_X^2 s_Y^2 s_Z^2 + c_X^2 s_Y^2 s_Z^2 - s_X^2 s_Z^2 + s_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z) c_X c_Z/(c'_Y)^2
\end{aligned}$$

$$+ (c_X^2 s_Z^2 + c_X^2 c_Z^2 + s_X^2 c_Z^2 - c_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z) s_X s_Y s_Z / (c'_Y)^2$$
$$= (c_Y^2 + s_Y^2 s_Z^2 - s_X^2 s_Z^2 + s_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z) c_X c_Z / (c'_Y)^2$$
$$+ (s_X^2 c_Z^2 + c_Z^2 - c_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z) s_X s_Y s_Z / (c'_Y)^2 \quad \text{as} \quad c_X^2 + s_X^2 = 1$$
$$= (1 - s_Y^2 + s_Y^2 (1 - c_Z^2) - s_X^2 s_Z^2 + (1 - c_X^2) s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z) c_X c_Z / (c'_Y)^2$$
$$+ (s_X^2 c_Z^2 + c_Z^2 - c_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z) s_X s_Y s_Z / (c'_Y)^2$$
$$= (c'_Y)^2 c_X c_Z / (c'_Y)^2 + (c'_Y)^2 s_X s_Y s_Z / (c'_Y)^2 \qquad \text{by (A-5) and (A-6)}$$
$$= c_X c_Z + s_X s_Y s_Z.$$
$$= r_{2,2}.$$

Proof of (A-3):

$$s'_X s'_Z - c'_X s'_Y c'_Z = (-r_{2,3}/c'_Y)(-r_{1,2}/c'_Y) - (r_{3,3}/c'_Y) r_{1,3} (r_{1,1}/c'_Y)$$
$$= [(s_X c_Z - c_X s_Y s_Z)(c_X s_Z - s_X s_Y c_Z) - c_X c_Y (s_X s_Z + c_X s_Y c_Z) c_Y c_Z] / (c'_Y)^2$$
$$= [s_X c_X s_Z c_Z - s_X^2 s_Y c_Z^2 - c_X^2 s_Y s_Z^2 + s_X c_X s_Y^2 s_Z c_Z - s_X c_X c_Y^2 s_Z c_Z - c_X^2 s_Y c_Y^2 c_Z^2] / (c'_Y)^2$$
$$= [s_X c_X s_Z c_Z - s_X^2 s_Y c_Z^2 - c_X^2 s_Y s_Z^2 + 2 s_X c_X s_Y^2 s_Z c_Z - s_X c_X s_Z c_Z - c_X^2 s_Y c_Y^2 c_Z^2] / (c'_Y)^2$$
$$\text{as} \quad s_Y^2 - c_Y^2 = 2 s_Y^2 - 1$$
$$= [-s_X^2 s_Y c_Z^2 - c_X^2 s_Y s_Z^2 + 2 s_X c_X s_Y^2 s_Z c_Z - c_X^2 s_Y (1 - s_Y^2) c_Z^2] / (c'_Y)^2$$
$$= [-s_X^2 s_Y c_Z^2 - c_X^2 s_Y s_Z^2 + 2 s_X c_X s_Y^2 s_Z c_Z - c_X^2 s_Y c_Z^2 + c_X^2 s_Y^3 c_Z^2] / (c'_Y)^2$$
$$= (-s_Y c_Z^2 - c_X^2 s_Y s_Z^2 + 2 s_X c_X s_Y^2 s_Z c_Z + c_X^2 s_Y^3 c_Z^2) / (c'_Y)^2 \quad \text{as} \quad -s_X^2 - c_X^2 = -1$$
$$= -s_Y \quad \text{by (A1.6)}$$
$$= r_{3,1}.$$

Proof of (A-4):

$$c'_X s'_Y s'_Z + s'_X c'_Z = (r_{3,3}/c'_Y) r_{1,3} (-r_{1,2}/c'_Y) + (-r_{2,3}/c'_Y)(r_{1,1}/c'_Y)$$
$$= [c_X c_Y (s_X s_Z + c_X s_Y c_Z)(c_X s_Z - s_X s_Y c_Z) + (s_X c_Z - c_X s_Y s_Z) c_Y c_Z] / (c'_Y)^2$$
$$= [(s_X c_X^2 s_Z^2 - s_X^2 c_X s_Y s_Z c_Z + c_X^3 s_Y s_Z c_Z - s_X c_X^2 s_Y^2 c_Z^2 + s_X c_Z^2 - c_X s_Y s_Z c_Z) c_Y] / (c'_Y)^2$$
$$= [(s_X c_X^2 s_Z^2 - s_X c_X^2 s_Y^2 c_Z^2 + c_X s_Y s_Z c_Z - 2 s_X^2 c_X s_Y s_Z c_Z + s_X c_Z^2 - c_X s_Y s_Z c_Z) c_Y] / (c'_Y)^2$$
$$\text{since} \quad -s_X^2 + c_X^2 = 1 - 2 s_X^2$$
$$= [(s_X c_X^2 s_Z^2 - s_X c_X^2 s_Y^2 c_Z^2 - 2 s_X^2 c_X s_Y s_Z c_Z + s_X c_Z^2) c_Y] / (c'_Y)^2$$
$$= s_X c_Y \quad \text{by (A1.6)}$$
$$= r_{3,2}.$$

A.2 Rotation-parameter conversion from Version 2 to Version 1

Section 2.6.2 describes how to convert Version-2 parameters R_X, R_Y and R_Z into Version-1 parameters R'_X, R'_Y and R'_Z which lead to the same rotation matrix. In so doing, it left the following identities unverified.

$$s'_X s'_Y c'_Z - c'_X s'_Z = r_{1,2}. \qquad \text{(A-7)}$$
$$s'_X s'_Z + c'_X s'_Y c'_Z = r_{1,3}. \qquad \text{(A-8)}$$
$$c'_X c'_Z + s'_X s'_Y s'_Z = r_{2,2}. \qquad \text{(A-9)}$$
$$c'_X s'_Y s'_Z - s'_X c'_Z = r_{2,3}. \qquad \text{(A-10)}$$

The proofs make use of all the equations in subsection 2.6.2 except for (2-30) because the above identities have still to be proved. They also use the trigonometrical identity $\sin^2\theta + \cos^2\theta = 1$ and its variations, notably $\cos^2\theta - \sin^2\theta = 1 - 2\sin^2\theta$ and $\sin^2\theta - \cos^2\theta = 1 - 2\cos^2\theta$.

The proofs all involve $(c'_Y)^2$ as a common factor in both parts of a quotient. This means it is assumed that c'_Y is non-zero. The special case where c'_Y is zero is covered separately in subsection 2.6.2.

From (2-32),

$$(c'_Y)^2 = 1 - s_X^2 s_Z^2 - c_X^2 s_Y^2 c_Z^2 + 2 s_X c_X s_Y s_Z c_Z. \qquad \text{(A-11)}$$

Since $1 - s_X^2 s_Z^2 = (s_X^2 + c_Z^2) - s_Z^2 (1 - c_X^2)$, equation (A-11) can be rewritten as

$$(c'_Y)^2 = c_X^2 + s_X^2 c_Z^2 - c_X^2 s_Y^2 c_Z^2 + 2 s_X c_X s_Y s_Z c_Z. \qquad \text{(A-12)}$$

Since $1 - s_X^2 s_Z^2 = (s_Z^2 + c_Z^2) - (1 - c_X^2) s_Z^2$, equation (A-11) can also be rewritten as

$$(c'_Y)^2 = c_Z^2 + c_X^2 s_Z^2 - c_X^2 s_Y^2 c_Z^2 + 2 s_X c_X s_Y s_Z c_Z. \qquad \text{(A-13)}$$

Proof of (A-7):

$$\begin{aligned}
s'_X s'_Y c'_Z - c'_X s'_Z &= (r_{3,2}/c'_Y)(-r_{3,1})(r_{1,1}/c'_Y) - (r_{3,3}/c'_Y)(r_{2,1}/c'_Y) \\
&= [(c_X s_Y s_Z + s_X c_Z)(c_X s_Y c_Z - s_X s_Z) c_Y c_Z - c_X c_Y (s_X s_Y c_Z + c_X s_Z)]/(c'_Y)^2 \\
&= [(c_X^2 s_Y^2 s_Z c_Z^2 - s_X c_X s_Y s_Z^2 c_Z + s_X c_X s_Y c_Z^3 - s_X^2 s_Z c_Z^2 - s_X c_X s_Y c_Z - c_X^2 s_Z) c_Y]/(c'_Y)^2 \\
&= [(c_X^2 s_Y^2 s_Z c_Z^2 - c_X^2 s_Z - s_X^2 s_Z c_Z^2 + s_X c_X s_Y c_Z^3 - s_X c_X s_Y c_Z - s_X c_X s_Y s_Z^2 c_Z) c_Y]/(c'_Y)^2 \\
&\qquad \text{by rearrangement of terms} \\
&= [(c_X^2 s_Y^2 s_Z c_Z^2 - c_X^2 s_Z - s_X^2 s_Z c_Z^2 + s_X c_X s_Y c_Z^3 - s_X c_X s_Y c_Z^3 - 2 s_X c_X s_Y s_Z^2 c_Z) c_Y]/(c'_Y)^2 \\
&\qquad \text{as } -1 - s_Z^2 = -c_Z^2 - 2 s_Z^2 \\
&= [(c_X^2 s_Y^2 s_Z c_Z^2 - c_X^2 s_Z - s_X^2 s_Z c_Z^2 - 2 s_X c_X s_Y s_Z^2 c_Z) c_Y]/(c'_Y)^2 \\
&= [-(c'_Y)^2 c_Y s_Z]/(c'_Y)^2 \quad \text{by (A.1.12)} \\
&= -c_Y s_Z \\
&= r_{1,2}.
\end{aligned}$$

Proof of (A-8):

$$\begin{aligned}
s'_X s'_Z + c'_X s'_Y c'_Z &= (r_{3,2}/c'_Y)(r_{2,1}/c'_Y) + (r_{3,3}/c'_Y)(-r_{3,1})(r_{1,1}/c'_Y) \\
&= [(c_X s_Y s_Z + s_X c_Z)(s_X s_Y c_Z + c_X s_Z) + c_X c_Y (c_X s_Y c_Z - s_X s_Z) c_Y c_Z]/(c'_Y)^2 \\
&= [s_X c_X s_Y^2 s_Z c_Z + c_X^2 s_Y s_Z^2 + s_X^2 s_Y c_Z^2 + s_X c_X s_Z c_Z + c_X^2 s_Y c_Y^2 c_Z^2 - s_X c_X c_Y^2 s_Z c_Z]/(c'_Y)^2 \\
&= [s_X c_X s_Y^2 s_Z c_Z - s_X c_X c_Y^2 s_Z c_Z + s_X c_X s_Z c_Z + c_X^2 s_Y s_Z^2 + s_X^2 s_Y c_Z^2 + c_X^2 s_Y c_Y^2 c_Z^2]/(c'_Y)^2 \\
&\qquad \nwarrow \text{by rearrangement of terms} \\
&= [2 s_X c_X s_Y^2 s_Z c_Z - s_X c_X s_Z c_Z + s_X c_X s_Z c_Z + c_X^2 s_Y s_Z^2 + s_X^2 s_Y c_Z^2 + c_X^2 s_Y c_Y^2 c_Z^2]/(c'_Y)^2 \\
&\qquad \nwarrow \text{since } s_Y^2 - c_Y^2 = 2 s_Y^2 - 1 \\
&= [2 s_X c_X s_Y^2 s_Z c_Z + c_X^2 s_Y s_Z^2 + (1 - c_X^2) s_Y c_Z^2 + c_X^2 s_Y (1 - s_Y^2) c_Z^2]/(c'_Y)^2 \\
&= [2 s_X c_X s_Y^2 s_Z c_Z - c_X^2 s_Y c_Z^2 + s_X^2 s_Y c_Z^2 + s_Y c_Z^2]/(c'_Y)^2 \\
&= s_Y (c'_Y)^2/(c'_Y)^2 \quad \text{by (A-13)} \\
&= s_Y \\
&= r_{1,3}.
\end{aligned}$$

Proof of (A-9):

$$\begin{aligned}
c'_X c'_Z + s'_X s'_Y s'_Z &= (r_{3,3}/c'_Y)(r_{1,1}/c'_Y) + (r_{3,2}/c'_Y)(-r_{1,3})(r_{2,1}/c'_Y) \\
&= [(c_X c_Y)(c_Y c_Z) + (c_X s_Y s_Z + s_X c_Z)(c_X s_Y c_Z - s_X s_Z)(s_X s_Y c_Z + c_X s_Z)]/(c'_Y)^2 \\
&= [c_X c_Y^2 c_Z + (c_X s_Y s_Z + s_X c_Z)(s_X c_X s_Y^2 c_Z^2 + c_X^2 s_Y s_Z c_Z - s_X^2 s_Y s_Z c_Z - s_X c_X s_Z^2)]/(c'_Y)^2 \\
&= [c_X c_Y^2 c_Z + s_X c_X^2 s_Y^3 s_Z c_Z^2 + c_X^3 s_Y^2 s_Z^2 c_Z - s_X^2 c_X s_Y^2 s_Z^2 c_Z - s_X c_X^2 s_Y s_Z^3 \\
&\qquad + s_X^2 c_X s_y^2 c_Z^3 + s_X c_X^2 s_Y s_Z c_Z^2 - s_X^3 s_Y s_Z c_Z^2 - s_X^2 c_X s_Z^2 c_Z]/(c'_Y)^2 \\
&= [c_X^3 s_Y^2 s_Z^2 c_Z + c_X c_Y^2 c_Z + s_X c_X^2 s_Y s_Z c_Z^2 - s_X^2 c_X s_Z^2 c_Z \\
&\qquad + s_X c_X^2 s_Y^3 s_Z c_Z^2 - s_X^3 s_Y s_Z c_Z^2 + (s_X^2 c_X s_Y^2 c_Z^3 - s_X^2 c_X s_Y^2 s_Z^2 c_Z) - s_X c_X^2 s_Y s_Z^3]/(c'_Y)^2 \\
&\qquad \text{by rearrangement of terms} \\
&= [c_X^3 s_Y^2 s_Z^2 c_Z + c_X c_Y^2 c_Z + s_X c_X^2 s_Y s_Z c_Z^2 - s_X^2 c_X s_Z^2 c_Z \\
&\qquad + s_X c_X^2 s_Y^3 s_Z c_Z^2 - s_X^3 s_Y s_Z c_Z^2 + (s_X^2 c_X s_Y^2 c_Z - 2 s_X^2 c_X s_Y^2 s_Z^2 c_Z) \\
&\qquad + (s_X c_X^2 s_Y s_Z c_Z^2 - s_X c_X^2 s_Y s_Z)]/(c'_Y)^2 \\
&\qquad \text{since } c_Z^2 - s_Z^2 = 1 - 2 s_Z^2 \text{ and } -s_Z^2 = c_Z^2 - 1 \\
&= [c_X^3 s_Y^2 s_Z^2 c_Z + c_X c_Y^2 c_Z + 2 s_X c_X^2 s_Y s_Z c_Z^2 + s_X^2 c_X s_Y^2 c_Z - s_X^2 c_X s_Z^2 c_Z \\
&\qquad + s_X c_X^2 s_Y^3 s_Z c_Z^2 - s_X^3 s_Y s_Z c_Z^2 - 2 s_X^2 c_X s_Y^2 s_Z^2 c_Z - s_X c_X^2 s_Y s_Z^3]/(c'_Y)^2 \\
&= c_X c_Z (c_X^2 s_Y^2 s_Z^2 + c_Y^2 + 2 s_X c_X s_Y s_Z c_Z + s_X^2 s_Y^2 - s_X^2 s_Z^2)/(c'_Y)^2 \\
&\qquad + (c_X^2 s_Y^2 c_Z^2 - s_X^2 c_Z^2 + 2 s_X c_X s_Y s_Z c_Z - c_X^2) s_X s_Y s_Z/(c'_Y)^2 \\
&= c_X c_Z (c_X^2 s_Y^2 (1 - c_Z^2) + c_Y^2 + s_X^2 s_Y^2 - s_X^2 s_Z^2 + 2 s_X c_X s_Y s_Z c_Z)/(c'_Y)^2 \\
&\qquad + (c_X^2 s_Y^2 c_Z^2 - s_X^2 c_Z^2 + 2 s_X c_X s_Y s_Z c_Z - c_X^2) s_X s_Y s_Z/(c'_Y)^2 \\
&\qquad = c_X c_Z (c_X^2 s_Y^2 - c_X^2 s_Y^2 c_Z^2 + c_Y^2 + s_X^2 s_Y^2 - s_X^2 s_Z^2 + 2 s_X c_X s_Y s_Z c_Z)/(c'_Y)^2 \\
&\qquad + (c_X^2 s_Y^2 c_Z^2 - s_X^2 c_Z^2 + 2 s_X c_X s_Y s_Z c_Z - c_X^2) s_X s_Y s_Z/(c'_Y)^2 \\
&= c_X c_Z (s_Y^2 - c_X^2 s_Y^2 c_Z^2 + c_Y^2 - s_X^2 s_Z^2 + 2 s_X c_X s_Y s_Z c_Z)/(c'_Y)^2 \\
&\qquad + (c_X^2 s_Y^2 c_Z^2 - s_X^2 c_Z^2 + 2 s_X c_X s_Y s_Z c_Z - c_X^2) s_X s_Y s_Z/(c'_Y)^2 \Big\} \text{ as } c_X^2 + s_X^2 = 1 \\
&= c_X c_Z (1 - c_X^2 s_Y^2 c_Z^2 - s_X^2 s_Z^2 + 2 s_X c_X s_Y s_Z c_Z)/(c'_Y)^2 \\
&\qquad + (c_X^2 s_Y^2 c_Z^2 - s_X^2 c_Z^2 + 2 s_X c_X s_Y s_Z c_Z - c_X^2) s_X s_Y s_Z/(c'_Y)^2 \Big\} \text{ as } s_Y^2 + c_Y^2 = 1 \\
&= c_X c_Z (c'_Y)^2/(c'_Y)^2 + (-c'_Y)^2 s_X s_Y s_Z/(c'_Y)^2 \quad \text{by (A-11) and (A-12)} \\
&= c_X c_Z - s_X s_Y s_Z \\
&= r_{2,2}.
\end{aligned}$$

Proof of (A-10):

$$c'_X s'_Y s'_Z - s'_X c'_Z = (r_{3,3}/c'_Y) r_{1,3} (r_{2,1}/c'_Y) - (r_{3,2}/c'_Y)(r_{1,1}/c'_Y)$$

$= [c_X c_Y (c_X s_Y c_Z - s_X s_Z)(s_X s_Y c_Z + c_X s_Z) - (c_X s_Y s_Z + s_X c_Z) c_Y c_Z] / (c'_Y)^2$
$= [(s_X c_X^2 s_Y^2 c_Z^2 + c_X^3 s_Y s_Z c_Z - s_X^2 c_X s_Y s_Z c_Z - s_X c_X^2 s_Z^2 - c_X s_Y s_Z c_Z - s_X c_Z^2) c_Y] / (c'_Y)^2$
$= [(s_X c_X^2 s_Y^2 c_Z^2 - c_X s_Y s_Z c_Z + (c_X^3 s_Y s_Z c_Z - s_X^2 c_X s_Y s_Z c_Z) - s_X c_X^2 s_Z^2 - s_X c_Z^2) c_Y] / (c'_Y)^2$
by rearrangement of terms
$= [(s_X c_X^2 s_Y^2 c_Z^2 - c_X s_Y s_Z c_Z + (c_X s_Y s_Z c_Z - 2 s_X^2 c_X s_Y s_Z c_Z) - s_X c_X^2 s_Z^2 - s_X c_Z^2) c_Y] / (c'_Y)^2$
as $c_X^2 - s_X^2 = 1 - 2 s_X^2$
$= [(s_X c_X^2 s_Y^2 c_Z^2 - 2 s_X^2 c_X s_Y s_Z c_Z - s_X c_X^2 s_Z^2 - s_X c_Z^2) c_Y] / (c'_Y)^2$
$= [(c_X^2 s_Y^2 c_Z^2 - 2 s_X c_X s_Y s_Z c_Z - c_X^2 s_Z^2 - c_Z^2) s_X c_Y] / (c'_Y)^2$
$= (c'_Y)^2 s_X c_Y / (c'_Y)^2$ by (A-13)
$= -s_X c_Y$
$= r_{2,3}$.

APPENDIX B: APPROXIMATE ELLIPSOIDAL DISTANCES

The distance between two points on the ellipsoid is generally taken to the length of the geodesic between the points. The geodesic is the shortest ellipsoidal arc. Sodano's 4th method is a process that evaluates the length of a geodesic with a high degree of accuracy. However, it lacks the simplicity of the Pythagorean formula which calculates distances on a projected plane.

One example of a simple method is a spherical-arc estimate which treats the latitudes as if they were reduced latitudes. The first version considered here, "Type 1", uses the geometric mean of the Equatorial radius and the polar radius:

$$d_{i,j} = \sqrt{ab}\ \arccos\big(\sin\phi_i \sin\phi_j + \cos\phi_i \cos\phi_j \cos(\lambda_i - \lambda_j)\big). \tag{B-1}$$

Formula (B-1) can give computational problems if the expression in brackets is very close to 1. One way to avoid this is to allow the contingency alternative

$$\left.\begin{aligned} d_{i,j} &= \sqrt{ab}\ \sqrt{(\phi_i - \phi_j)^2 + (\lambda_i - \lambda_j)^2 \cos^2\phi_i} \\ &\text{if } \max(|\phi_i - \phi_j|, |\lambda_i - \lambda_j|) < 0.0000005. \end{aligned}\right\} \tag{B-2}$$

The following tables were obtained from the Type-1 method for lines varying in length from 500m to 100km. The range of mid-latitudes and the range of azimuths are sufficiently general because of the symmetry of the ellipsoid.

Percentage Errors In Type-1 Spherical-Arc Estimates for different azimuths and mid-latitudes on WGS84

	000°	010°	020°	030°	040°	050°	060°	070°	080°	090°
80°N	-0.47	-0.47	-0.47	-0.48	-0.48	-0.48	-0.49	-0.49	-0.49	-0.49
70°N	-0.38	-0.39	-0.39	-0.40	-0.42	-0.43	-0.44	-0.45	-0.46	-0.46
60°N	-0.25	-0.26	-0.27	-0.29	-0.32	-0.35	-0.38	-0.40	-0.41	-0.42
50°N	-0.09	-0.10	-0.12	-0.16	-0.20	-0.25	-0.29	-0.33	-0.36	-0.36
40°N	0.09	0.08	0.04	-0.01	-0.08	-0.14	-0.21	-0.26	-0.29	-0.31
30°N	0.25	0.24	0.19	0.13	0.04	-0.04	-0.13	-0.19	-0.24	-0.25
20°N	0.39	0.37	0.32	0.24	0.14	0.04	-0.06	-0.14	-0.19	-0.21
10°N	0.47	0.45	0.40	0.31	0.20	0.09	-0.02	-0.10	-0.16	-0.18
Equator	0.51	0.48	0.43	0.34	0.23	0.11	0.00	-0.09	-0.15	-0.17
(This table is the same - with barely perceptible variations - for distances 500m, 1km, 2km, 5 km, 10km, 20km, 50km, 100km, 200km, 500km and 1000km.)										

Percentage Errors In Type-1 Spherical-Arc Estimates for different azimuths and mid-latitudes on Clarke 1880 Modified (where $f = 1/293.465$)

	000°	010°	020°	030°	040°	050°	060°	070°	080°	090°
80°N	-0.48	-0.48	-0.48	-0.48	-0.49	-0.49	-0.50	-0.50	-0.50	-0.50
70°N	-0.39	-0.39	-0.40	-0.41	-0.42	-0.44	-0.45	-0.46	-0.47	-0.47
60°N	-0.26	-0.26	-0.28	-0.30	-0.33	-0.36	-0.38	-0.41	-0.42	-0.43
50°N	-0.09	-0.10	-0.12	-0.16	-0.20	-0.25	-0.30	-0.34	-0.36	-0.37
40°N	0.09	0.08	0.04	-0.01	-0.08	-0.15	-0.21	-0.26	-0.30	-0.31
30°N	0.26	0.24	0.20	0.13	0.04	-0.04	-0.13	-0.20	-0.24	-0.26
20°N	0.39	0.37	0.32	0.24	0.14	0.04	-0.06	-0.14	-0.19	-0.21
10°N	0.48	0.46	0.40	0.32	0.21	0.09	-0.02	-0.10	-0.16	-0.18
Equator	0.51	0.49	0.43	0.34	0.23	0.11	0.00	-0.09	-0.15	-0.17
(This table is the same - with barely perceptible variations - for distances 500m, 1km, 2km, 5 km, 10km, 20km, 50km, 100km, 200km, 500km and 1000km.)										

Percentage Errors In Type-1 Spherical-Arc Estimates for different azimuths and mid-latitudes on Krayenhoff 1827 (where $f = 1/309.65$)

	000°	010°	020°	030°	040°	050°	060°	070°	080°	090°
80°N	-0.45	-0.46	-0.46	-0.46	-0.46	-0.47	-0.47	-0.47	-0.47	-0.47
70°N	-0.37	-0.37	-0.38	-0.39	-0.40	-0.42	-0.43	-0.44	-0.44	-0.45
60°N	-0.24	-0.25	-0.26	-0.28	-0.31	-0.34	-0.36	-0.38	-0.40	-0.40
50°N	-0.08	-0.09	-0.11	-0.15	-0.19	-0.24	-0.28	-0.32	-0.34	-0.35
40°N	0.09	0.07	0.04	-0.01	-0.07	-0.14	-0.20	-0.25	-0.28	-0.29
30°N	0.24	0.23	0.19	0.12	0.04	-0.04	-0.12	-0.19	-0.23	-0.24
20°N	0.37	0.36	0.31	0.23	0.14	0.04	-0.06	-0.13	-0.18	-0.20
10°N	0.46	0.44	0.38	0.30	0.20	0.09	-0.02	-0.10	-0.15	-0.17
Equator	0.49	0.47	0.41	0.32	0.22	0.10	0.00	-0.09	-0.14	-0.16
(This table is the same - with barely perceptible variations - for distances 500m, 1km, 2km, 5 km, 10km, 20km, 50km, 100km, 200km, 500km and 1000km.)										

The second version of spherical-arc estimate, "Type 2", uses the geometric mean of the radii of curvature at the line's mid-latitude:

$$d_{i,j} = \frac{a\sqrt{1-e^2}}{1-e^2\sin^2\phi_m}\ \arccos\big(\sin\phi_i\sin\phi_j + \cos\phi_i\cos\phi_j\cos(\lambda_i-\lambda_j)\big), \tag{B-3}$$

where

$$\phi_m = (\phi_i + \phi_j)/2. \tag{B-4}$$

Formula (B-3) can give computational problems if the expression in brackets is very close to 1. One way to avoid this is to allow the contingency alternative

$$\left.\begin{array}{l} d_{i,j} = \dfrac{a\sqrt{1-e^2}}{1-e^2\sin^2\phi_m}\sqrt{(\phi_i-\phi_j)^2 + (\lambda_i-\lambda_j)^2\cos^2\phi_i} \\ \text{if } \max(|\phi_i-\phi_j|, |\lambda_i-\lambda_j|) < 0.0000005. \end{array}\right\} \tag{B-5}$$

Percentage Errors in Type-2 Spherical-Arc Estimates for different azimuths and mid-latitudes on WGS84

	000°	010°	020°	030°	040°	050°	060°	070°	080°	090°
80°N	0.01	0.01	0.01	0.01	0.00	0.00	-0.01	-0.01	-0.01	-0.01
70°N	0.04	0.04	0.03	0.02	0.01	-0.01	-0.02	-0.03	-0.04	-0.04
60°N	0.08	0.08	0.06	0.04	0.01	-0.01	-0.04	-0.06	-0.08	-0.08
50°N	0.14	0.13	0.11	0.07	0.02	-0.02	-0.07	-0.11	-0.13	-0.14
40°N	0.20	0.19	0.15	0.10	0.03	-0.03	-0.10	-0.15	-0.19	-0.20
30°N	0.25	0.24	0.19	0.13	0.04	-0.04	-0.13	-0.19	-0.24	-0.25
20°N	0.30	0.28	0.23	0.15	0.05	-0.05	-0.15	-0.23	-0.28	-0.30
10°N	0.33	0.31	0.25	0.16	0.06	-0.06	-0.16	-0.25	-0.31	-0.33
Equator	0.34	0.32	0.26	0.17	0.06	-0.06	-0.17	-0.26	-0.32	-0.34
(This table is the same for distances 500m, 1km, 2km, 5 km, 10km, 20km, 50km, 100km, 200km, 500km and 1000km.)										

Percentage Errors In Type-2 Spherical-Arc Estimates for different azimuths and mid-latitudes on Clarke 1880 Modified (where $f = 1/293.465$)

	000°	010°	020°	030°	040°	050°	060°	070°	080°	090°
80°N	0.01	0.01	0.01	0.01	0.00	0.00	-0.01	-0.01	-0.01	-0.01
70°N	0.04	0.04	0.03	0.02	0.01	-0.01	-0.02	-0.03	-0.04	-0.04
60°N	0.09	0.08	0.07	0.04	0.01	-0.01	-0.04	-0.07	-0.08	-0.09
50°N	0.14	0.13	0.11	0.07	0.02	-0.02	-0.07	-0.11	-0.13	-0.14
40°N	0.20	0.19	0.15	0.10	0.03	-0.04	-0.10	-0.15	-0.19	-0.20
30°N	0.26	0.24	0.20	0.13	0.04	-0.05	-0.13	-0.20	-0.24	-0.26
20°N	0.30	0.28	0.23	0.15	0.05	-0.05	-0.15	-0.23	-0.28	-0.30
10°N	0.33	0.31	0.25	0.16	0.06	-0.06	-0.17	-0.25	-0.31	-0.33
Equator	0.34	0.32	0.26	0.17	0.06	-0.06	-0.17	-0.26	-0.32	-0.34

(This table is the same for distances 500m, 1km, 2km, 5 km, 10km, 20km, 50km, 100km, 200km, 500km and 1000km.)

Percentage Errors In Type-2 Spherical-Arc Estimates for different azimuths and mid-latitudes on Krayenhoff 1827 (where $f = 1/309.65$)

	000°	010°	020°	030°	040°	050°	060°	070°	080°	090°
80°N	0.01	0.01	0.01	0.00	0.00	0.00	0.00	-0.01	-0.01	-0.01
70°N	0.04	0.04	0.03	0.02	0.01	-0.01	-0.02	-0.03	-0.04	-0.04
60°N	0.08	0.08	0.06	0.04	0.01	-0.01	-0.04	-0.06	-0.08	-0.08
50°N	0.13	0.13	0.10	0.07	0.02	-0.02	-0.07	-0.10	-0.13	-0.13
40°N	0.19	0.18	0.15	0.09	0.03	-0.03	-0.10	-0.15	-0.18	-0.19
30°N	0.24	0.23	0.19	0.12	0.04	-0.04	-0.12	-0.19	-0.23	-0.24
20°N	0.29	0.27	0.22	0.14	0.05	-0.05	-0.14	-0.22	-0.27	-0.29
10°N	0.31	0.30	0.24	0.16	0.05	-0.06	-0.16	-0.24	-0.29	-0.31
Equator	0.32	0.30	0.25	0.16	0.06	-0.06	-0.16	-0.25	-0.30	-0.32
(This table is the same for distances 500m, 1km, 2km, 5 km, 10km, 20km, 50km, 100km, 200km, 500km and 1000km.)										

Another method is to use an ellipsoidal chord approximation corrected to an arc. Taking h as zero, the Cartesian coordinates corresponding to (ϕ_i, λ_i) are computed as follows.

$$\nu_i = \frac{a}{\sqrt{1-e^2 \sin^2 \phi_i}}. \tag{B-6}$$

$$X_i = \nu_i \cos\phi_i \cos\lambda_i. \tag{B-7}$$

$$Y_i = \nu_i \cos\phi_i \sin\lambda_i. \tag{B-8}$$

$$Z_i = \nu_i (1 - e^2) \sin \phi_i. \tag{B-9}$$

The ellipsoidal-chord approximation to the geodesic is:

$$c_{i,j} = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2 + (Z_j - Z_i)^2}. \tag{B-10}$$

This is corrected from a chord-approximation to an arc-approximation as follows:

$$d_{i,j} = c_{i,j} + \frac{c_{i,j}^3}{24R^2} + \frac{3c_{i,j}^5}{640R^4} \tag{B-11}$$

where R is a suitable estimate of the radius.

This is based on the fact that a chord c of a circle with radius R which subtends an angle of α radians satisfies

$$\alpha = 2\sin^{-1}\frac{c}{2R} \approx 2\left(\frac{c}{2R} + \frac{c^3}{48R^3} + \frac{3c^5}{1280R^5}\right), \tag{B-12}$$

which in turn leads to

$$R\alpha \approx c + \frac{c^3}{24R^2} + \frac{3c^5}{640R^4}. \tag{B-13}$$

The most suitable estimate of R appears to be $\sqrt[4]{\rho_i \nu_i \rho_j \nu_j}$, a geometric mean of 4 radii of curvature. The formula for $d_{i,j}$ only requires the evaluation of

$$R^2 = \frac{a^2(1-e^2)}{(1-e^2 \sin^2 \phi_i)(1-e^2 \sin^2 \phi_j)}. \tag{B-14}$$

The error was less than 1 in 141000 for lines up to 1000km long. The largest errors occurred near the Equator when the azimuth was near 000° or 090°.

Percentage Errors In Chord-To-Arc Estimates for different distances

Geodesic Length	WGS84	Clarke 1880 Modified	Krayenhoff 1827
1000km	-0.00070 to 0.00070	-0.00071 to 0.00071	-0.00067 to 0.00067
500km	-0.00017 to 0.00018	-0.00018 to 0.00018	-0.00017 to 0.00017
200km	-0.00003 to 0.00003	-0.00003 to 0.00003	-0.00003 to 0.00003
100km	-0.00001 to 0.00001	-0.00001 to 0.00001	-0.00001 to 0.00001
50km	0.00000	0.00000	0.00000
20km	0.00000	0.00000	0.00000
10km	0.00000	0.00000	0.00000
5km	0.00000	0.00000	0.00000
2km	0.00000	0.00000	0.00000
1km	0.00000	0.00000	0.00000
500m	0.00000	0.00000	0.00000
Mid-latitudes covered were Equator, 10°N, 20°N, ... , 80°N. Azimuths covered were 000°, 010°, 020°, ... , 090°.			

Details of the worst case are shown in the following table.

Percentage Errors In Chord-To-Arc Estimates for different azimuths and mid-latitudes on Clarke 1880 Modified over 1000km

	000°	010°	020°	030°	040°	050°	060°	070°	080°	090°
80°N	-0.00001	-0.00001	-0.00001	-0.00001	0.00000	0.00001	0.00001	0.00002	0.00002	0.00002
70°N	-0.00008	-0.00007	-0.00006	-0.00004	-0.00001	0.00002	0.00004	0.00006	0.00008	0.00008
60°N	-0.00017	-0.00016	-0.00013	-0.00009	-0.00003	0.00003	0.00009	0.00014	0.00017	0.00018
50°N	-0.00029	-0.00027	-0.00022	-0.00015	-0.00005	0.00005	0.00015	0.00022	0.00027	0.00029
40°N	-0.00041	-0.00039	-0.00032	-0.00021	-0.00007	0.00007	0.00021	0.00032	0.00039	0.00041
30°N	-0.00053	-0.00050	-0.00041	-0.00027	-0.00009	0.00009	0.00026	0.00041	0.00050	0.00053
20°N	-0.00063	-0.00059	-0.00048	-0.00031	-0.00011	0.00011	0.00031	0.00048	0.00059	0.00062
10°N	-0.00069	-0.00065	-0.00053	-0.00034	-0.00012	0.00012	0.00034	0.00053	0.00065	0.00069
Equator	-0.00071	-0.00067	-0.00054	-0.00035	-0.00012	0.00012	0.00035	0.00054	0.00067	0.00071

This appendix describes actual datasets of control points in specific datums in actual parts of the world. In some cases the ellipsoidal heights have been generated artificially and this is noted where applicable. Where the datasets contain less than 100 points, their coordinates in the respective datums are given.

C.1 Western Australia (AGD84, GDA94)

This dataset consists of 82 points from the STATEFIX GPS network. The coordinates are known in Australian Geodetic Datum 1984 (AGD84) and the Geocentric Datum of Australia 1994 (GDA94). The former is based on the Australia National (1966) ellipsoid. The latter is a realisation of ITRF92 at Epoch 1994.0, so its ellipsoid is GRS80. The data was provided by Joseph Awange. The distribution of the data points is shown in Figure C-1.

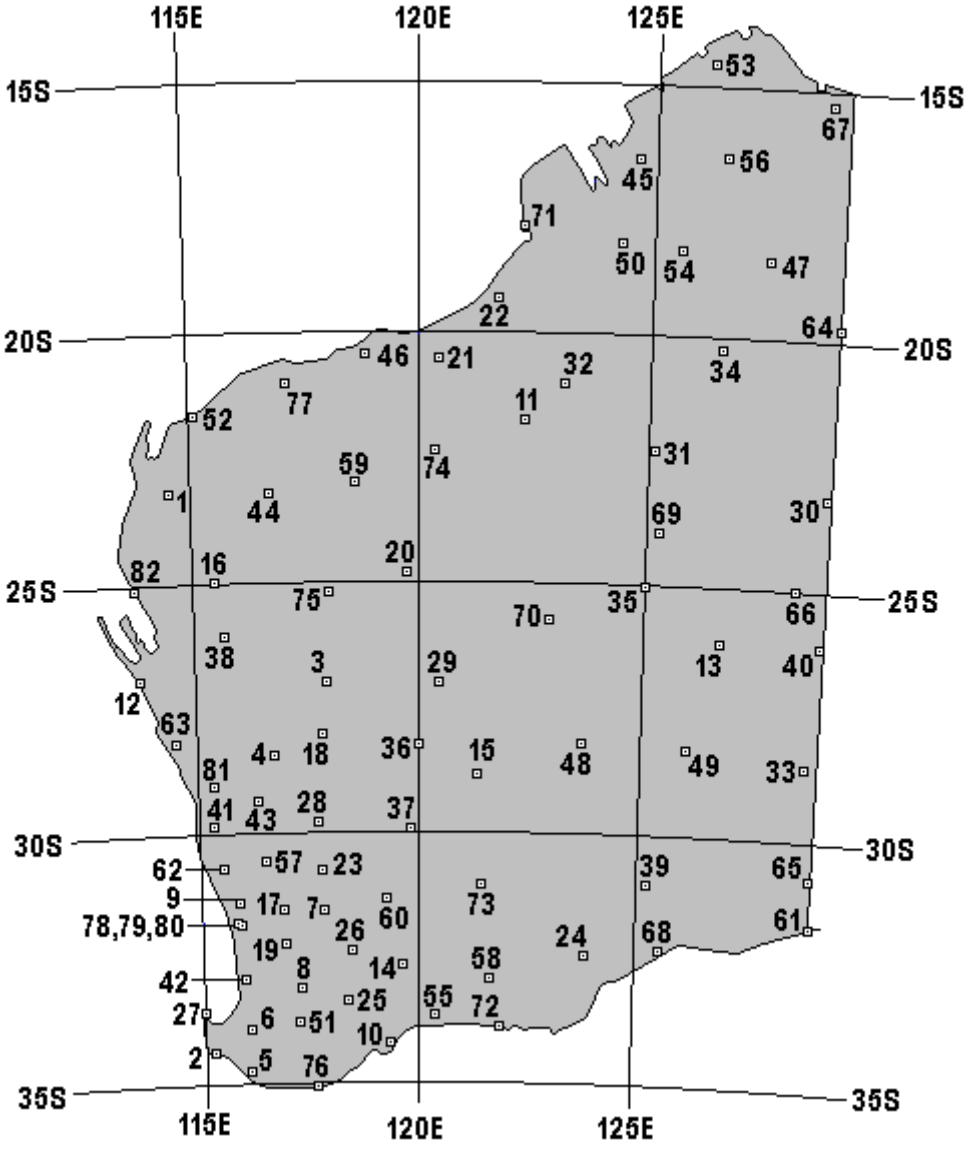


Figure C-1: Data points for Western Australia, where the numbers are extracted from the unofficial point identifications ACP01 to ACP82.

The geodetic coordinates of the common points are as follows.

Unofficial	AGD84 Coordinates			World Geodetic System 1984 Coordinates		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)	Latitude (deg)	Longitude (deg)	Ell ht (m)
ACP01	-23.150590580	114.505187133	116.0108	-23.149301629	114.506533608	109.0597

ACP02	-34.375429357	115.135167915	39.4521	-34.374235748	115.136692177	8.4468
ACP03	-26.990289655	117.931712036	599.5410	-26.989001238	117.933116841	590.1900
ACP04	-28.437118502	116.726719563	456.0513	-28.435855706	116.728147005	441.0406
ACP05	-34.766049299	116.085340159	207.8515	-34.764847453	116.086867385	177.6619
ACP06	-33.923955138	116.117779195	295.2417	-33.922744939	116.119288468	266.8849
ACP07	-31.579587583	117.844980684	345.9015	-31.578337969	117.846448282	326.0284
ACP08	-33.063953927	117.257439469	345.8114	-33.062723795	117.258931622	321.5060
ACP09	-31.395142498	115.930103377	224.9517	-31.393911393	115.931576249	202.0698
ACP10	-34.195108938	119.316693083	199.0415	-34.193867944	119.318192405	175.9780
ACP11	-21.764166676	122.316262169	346.2313	-21.762788216	122.317586057	358.3130
ACP12	-26.906764101	113.778514386	298.2707	-26.905518657	113.779921296	281.1357
ACP13	-26.113771343	126.665187256	628.2310	-26.112380837	126.666545538	638.0695
ACP14	-32.633129057	119.611540531	448.3912	-32.631869023	119.613014790	429.3052
ACP15	-28.820158433	121.299573168	469.0808	-28.818844913	121.300991852	462.6592
ACP16	-24.964624574	115.468866144	265.7712	-24.963336407	115.470243910	256.6562
ACP17	-31.526000950	116.893989478	315.1118	-31.524760950	116.895460374	293.8022
ACP18	-28.032610530	117.823331698	561.3113	-28.031330262	117.824746729	549.6412
ACP19	-32.202589801	116.918407412	318.4618	-32.201355497	116.919887870	295.5785
ACP20	-24.818019181	119.762822864	783.1616	-24.816691655	119.764196840	783.2239
ACP21	-20.521946504	120.416845308	270.8003	-20.520571944	120.418167654	282.2376
ACP22	-19.316137538	121.713341001	109.5208	-19.314755821	121.714650045	126.4676
ACP23	-30.765637939	117.781053335	424.3516	-30.764382413	117.782510257	406.2987
ACP24	-32.449371189	123.805711860	176.2020	-32.448062862	123.807180515	165.6021
ACP25	-33.343448382	118.342434137	389.4314	-33.342209439	118.343925160	366.4546
ACP26	-32.351601097	118.471796492	401.6017	-32.350351462	118.473272300	381.1755
ACP27	-33.575456919	115.023532923	213.4721	-33.574257011	115.025041876	184.0037
ACP28	-29.777086494	117.702996054	408.8215	-29.775822964	117.704438796	392.8399
ACP29	-26.992611554	120.465982828	566.0715	-26.991285886	120.467388026	562.0398
ACP30	-23.153863096	128.899263830	552.3114	-23.152435773	128.900567883	574.1401
ACP31	-22.313015705	125.115127411	371.0309	-22.311619990	125.116451116	387.6423
ACP32	-20.985807011	123.182878152	287.5008	-20.984424976	123.184194550	303.7394
ACP33	-28.519861171	128.735804144	329.8411	-28.518465434	128.737169276	337.1649
ACP34	-20.261119781	126.540629721	432.2208	-20.259708145	126.541922189	457.1863
ACP35	-25.050599490	124.982459997	556.5811	-25.049214943	124.983815816	566.1066
ACP36	-28.236489121	119.977347764	476.8512	-28.235180928	119.978767271	468.8738
ACP37	-29.810740072	119.916113744	524.0816	-29.809451527	119.917546755	512.3317
ACP38	-26.052999288	115.669261133	344.7112	-26.051722873	115.670649141	333.2244
ACP39	-31.019834658	125.247033553	195.0620	-31.018495014	125.248459027	190.0200
ACP40	-26.128083514	128.929402057	1025.4514	-26.126670829	128.930737375	1039.5300
ACP41	-29.833135366	115.333069015	306.0416	-29.831897495	115.334521239	285.5629
ACP42	-32.939014132	116.022679002	501.9017	-32.937796310	116.024174606	475.8754
ACP43	-29.366272500	116.375170544	370.0515	-29.365020120	116.376612810	352.4156
ACP44	-23.189343846	116.700031788	243.0006	-23.188040308	116.701377899	240.8377
ACP45	-16.473879929	124.619095734	489.0804	-16.472466823	124.620371640	519.0436
ACP46	-20.438913298	118.842826416	108.35089	-20.437558564	118.844152153	116.8184
ACP47	-18.428735304	127.423894356	467.91039	-18.427308006	127.425161840	499.4107
ACP48	-28.210244020	123.655614610	518.1215	-28.208895614	123.657019759	517.2738
ACP49	-28.290009547	126.046473194	387.80186	-28.288636903	126.047859236	390.8686
ACP50	-18.147970276	124.300625474	107.37107	-18.146564587	124.301915531	133.0391
ACP51	-33.743257823	117.228693522	401.96161	-33.742034647	117.230195616	376.0809
ACP52	-21.635809035	115.107218698	33.481098	-21.634496804	115.108554519	31.3304
ACP53	-14.569450474	126.135038230	335.30006	-14.568025753	126.136300864	373.4775
ACP54	-18.291633496	125.587334282	168.77096	-18.290219232	125.588618255	196.7109
ACP55	-33.644122194	120.375253949	241.31168	-33.642862454	120.376739724	221.5023
ACP56	-16.419966161	126.438335668	527.33999	-16.418544556	126.439606241	561.5712
ACP57	-30.559107570	116.509589719	344.37097	-30.557863234	116.511048253	324.4127
ACP58	-32.923763834	121.641857150	320.69106	-32.922479462	121.643326029	304.9110
ACP59	-22.981808152	118.586608986	1264.4314	-22.980481390	118.587958503	1266.5445
ACP60	-31.331920275	119.278571392	481.63144	-31.330651944	119.280029075	465.1484
ACP61	-31.686659477	129.011285739	90.731733	-31.685284813	129.012695854	90.7589
ACP62	-30.671800918	115.510645422	237.85201	-30.670568518	115.512109204	216.0141
ACP63	-28.180942809	114.508607883	291.88153	-28.179699786	114.510043193	273.1032
ACP64	-19.768614366	128.978239361	487.71043	-19.767175183	128.979516008	519.3603
ACP65	-30.771659519	128.953964921	164.62174	-30.770274788	128.955360266	166.6600

ACP66	-24.996162901	128.311317336	832.31056	-24.994749862	128.312644684	848.1381
ACP67	-15.607117612	128.275121305	68.820302	-15.605679381	128.276370712	109.0092
ACP68	-32.264919799	125.536248528	124.0612	-32.263590200	125.537695185	116.6583
ACP69	-23.945679703	125.243597420	424.70084	-23.944288391	125.244937525	436.5400
ACP70	-25.708455378	122.908253052	499.67111	-25.707092871	122.909625219	503.1036
ACP71	-17.890768806	122.265926364	29.410184	-17.889379522	122.267226435	50.8267
ACP72	-33.875478795	121.893116555	69.442127	-33.874202692	121.894588651	51.7895
ACP73	-31.056626939	121.447645895	480.98162	-31.055328176	121.449095588	469.3139
ACP74	-22.447935483	120.340290194	548.93119	-22.446574547	120.341628765	555.6634
ACP75	-25.215554957	118.005606053	496.14137	-25.214247742	118.006985906	491.4397
ACP76	-35.079549935	117.619893862	288.76181	-35.078334769	117.621417859	260.7850
ACP77	-20.982769624	117.095852305	128.2405	-20.981432571	117.097186821	132.1709
ACP78	-31.841415187	115.974061251	40.021347	-31.840187871	115.975540551	16.2867
ACP79	-31.803193242	115.883766836	59.541223	-31.801966629	115.885246131	35.6873
ACP80	-31.802392826	115.883697472	59.791836	-31.801166039	115.885176632	35.9319
ACP81	-29.047800745	115.345530331	283.2913	-29.046555740	115.346972361	264.1519
ACP82	-25.115263429	113.730181558	28.651094	-25.113995688	113.731558211	15.5439

C.2 Great Britain (OSGB36, WGS84)

This dataset consists of 44 points known in Ordnance Survey Great Britain 1936 (OSGB36) and WGS84. The former is based on the Airy ellipsoid. The data was supplied by E. J. Price, who obtained it from Military contacts circa 2001, but it is believed to have originated from Ordnance Survey. The distribution of the data points is shown in Figure C-2.

1	20280	23	31036
2	30118	24	31037
3	30210	25	31092
4	30219	26	31093
5	30229	27	31094
6	30230	28	31099
7	30231	29	31147
8	30438	30	31148
9	30729	31	31149
10	30737	32	80046
11	30738	33	80054
12	30739	34	80209
13	30753	35	80210
14	30754	36	80211
15	30755	37	80212
16	30760	38	80215
17	30770	39	80305
18	30771	40	80306
19	30772	41	80307
20	30773	42	80308
21	30938	43	80309
22	31035	44	80335

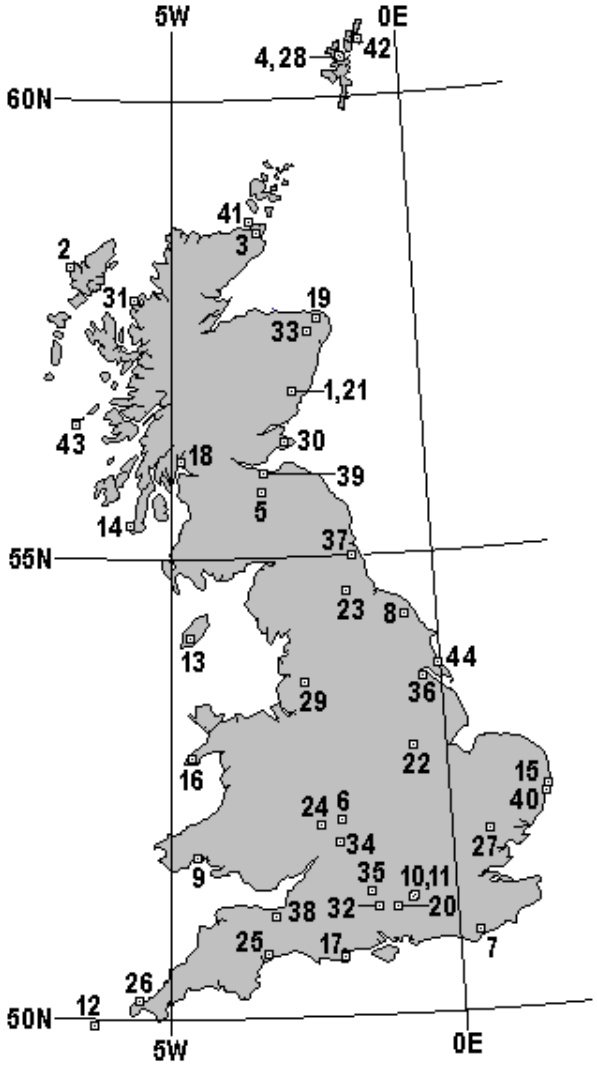


Figure C-2: Data points for Great Britain, cross-referenced to official point identifications.

The geodetic coordinates of the common points are as follows.

	OSGB36 Coordinates			World Geodetic System 1984 Coordinates		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)	Latitude (deg)	Longitude (deg)	Ell ht (m)
20280	56.811210560	-2.607177223	46.4000	56.811063889	-2.608715833	96.8100
30118	58.186534830	-7.096123303	41.2000	58.186118889	-7.097141944	96.5800
30210	58.552583080	-3.240778028	53.7700	58.552196667	-3.242323056	103.7000
30219	60.439030080	-1.299205417	12.6600	60.438475556	-1.301152222	60.5100
30229	55.733752500	-3.227309472	281.1000	55.733719167	-3.228731389	332.3800
30230	52.142984780	-1.966633639	113.1700	52.143400556	-1.968059167	160.7000
30231	50.865353140	0.346078195	33.0500	50.865936389	0.344472778	76.3700
30438	54.355322220	-0.666935279	256.6200	54.355506111	-0.668646111	303.6100
30729	51.758584560	-4.558782389	185.9800	51.759019722	-4.559921667	238.7900
30737	51.273893890	-0.771001389	67.9400	51.274414167	-0.772528333	112.0500
30738	51.286577220	-0.754109722	75.6600	51.287101389	-0.755629722	120.5700
30739	49.923830050	-6.280106721	37.6800	49.924440278	-6.281050556	89.1800
30753	54.149324970	-4.668137777	483.7200	54.149480000	-4.669317778	537.5300
30754	55.372798530	-5.770368889	446.8000	55.372778889	-5.771415556	500.2500
30755	52.415485190	1.717449223	46.6100	52.415941111	1.715609444	90.1900
30760	52.832522610	-4.630083722	304.8500	52.832839444	-4.631225556	358.1800
30770	50.629786750	-1.988219833	199.6600	50.630360278	-1.989581389	245.9400
30771	56.058331940	-4.816235278	22.2100	56.058229444	-4.817457778	76.2600
30772	57.602849440	-2.031349445	230.6800	57.602601944	-2.033006944	278.7400
30773	51.185069440	-1.025637778	186.4500	51.185597222	-1.027138333	232.0400
30938	56.812213060	-2.598661945	51.1500	56.812049722	-2.600231944	100.8100
31035	52.910665690	-0.592800556	129.0800	52.911027500	-0.594424722	176.2900
31036	54.619261060	-1.683660028	222.5100	54.619388889	-1.685217778	273.6700
31037	52.104399560	-2.337563111	425.3000	52.104816944	-2.338935278	474.9700
31092	50.684010890	-3.360757389	176.4400	50.684564167	-3.361994444	226.7100
31093	50.194443920	-5.532992583	247.7400	50.195022778	-5.534014444	299.7400
31094	51.978086330	0.665113333	89.1100	51.978564444	0.663409167	132.9700
31099	60.436073360	-1.201255278	139.4100	60.435522778	-1.203225833	187.6800
31147	53.629588110	-2.514467083	456.8100	53.629833611	-2.515888056	508.8000
31148	56.248262250	-2.778813389	182.4000	56.248168611	-2.780323333	233.8500
31149	57.835966330	-5.765483945	296.6600	57.835625000	-5.766658056	352.6700
80046	51.184065560	-1.379804444	78.9300	51.184593056	-1.381260278	123.8900
80054	57.469287860	-2.212462861	177.4500	57.469059167	-2.214110000	226.9100
80209	51.919424640	-2.004448639	330.3500	51.919874444	-2.005873611	376.7700
80210	51.351970310	-1.463654778	297.1300	51.352483056	-1.465105000	343.6500
80211	53.675721330	-0.331394278	34.1000	53.675996389	-0.333100000	79.8200
80212	54.990480920	-1.531728972	51.7500	54.990572500	-1.533351111	99.3800
80215	51.109074940	-3.192448639	384.4900	51.109588333	-3.193725556	434.8700
80305	55.924035420	-3.177756389	94.6800	55.923974722	-3.179190556	147.0700
80306	52.327806110	1.680256416	41.5900	52.328271944	1.678439444	86.1700
80307	58.669491140	-3.370399027	127.3300	58.669090000	-3.371929722	178.0600
80308	60.620814670	-0.862825973	158.7100	60.620252778	-0.864840833	207.4200
80309	56.455486780	-6.922482971	141.0300	56.455303611	-6.923412500	197.3900
80335	53.783758530	-0.029333028	25.7600	53.784031389	-0.031075556	70.7800

C.3 Ghana's Golden Triangle (Accra, WGS84)

This dataset consists of 19 points known in the Accra datum (based on the War Office ellipsoid) and WGS84. Apart from the War Office ellipsoidal heights, the data comes from Dzidefo (2011). The points are located in Ghana's "Golden Triangle". This consists of five of the ten regions in Ghana, namely Ashanti, Greater Accra, Western, Central and Eastern (Ziggah *et al*, 2017). The distribution of the data points is shown in Figure C-3.

The War Office ellipsoidal heights are listed in Laari *et al* (2016). This source makes it clear that they were generated mathematically. Computations carried out for this study established

that the War Office heights are transformed to WGS84 heights exactly by the Abridged Molodensky transformation with parameters -196.63555, 33.360366, 322.518564. Those parameters have not been found in any publication, but the method that was actually used must have been equivalent.

The geodetic coordinates are as follows.

	Accra Coordinates			World Geodetic System 1984 Coordinates		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)	Latitude (deg)	Longitude (deg)	Ell ht (m)
CFP 109	5.457304069	-0.423846053	82.0662	5.460090469	-0.423560461	78.2744
CFP 200	5.623015028	-0.559597439	307.9119	5.625798375	-0.559316989	304.9379
CFP 225	5.452274171	-1.501354089	279.5252	5.455086786	-1.501101706	275.1437
CFP 155	5.936246667	-0.122290556	525.5954	5.939034008	-0.121995097	524.5492
GCS 179	6.369342009	-1.033574469	491.7309	6.372117933	-1.033307164	492.5083
GCS 142	6.573034939	-0.765843033	780.2024	6.575796806	-0.765572008	782.2084
CFP 213	6.125418053	-0.749175303	327.4004	6.128197050	-0.748904353	327.0218
CFP 178	6.568591978	-1.164931611	613.9824	6.571357297	-1.164662747	615.7568
CFP 306	7.233128636	-1.630716606	530.9264	7.235861019	-1.630465086	536.0062
GCS 145R	6.554147306	-1.412160561	502.1391	6.556916419	-1.411897078	503.7124
CFP 207	5.846824211	-1.966403494	401.8272	5.849618067	-1.966153267	399.3477
CFP 217	5.940330581	-0.729977811	312.4534	5.943106975	-0.729704947	311.0926
GCS 302	6.909735047	-2.017006497	557.6481	6.912480178	-2.016757894	560.8285
CFP 304	6.989461464	-1.445590708	617.0558	6.992208703	-1.445337764	620.9316
CFP 305	6.843594833	-1.743673061	414.0673	6.846343694	-1.743417842	417.0231
CFP 184	6.468790536	-1.695085044	471.1630	6.471557664	-1.694831050	472.1430
GCS 102	5.279958350	-0.734677778	88.3693	5.282744181	-0.734406006	83.4515
CFP 180R	6.051006676	-1.286464464	438.7549	6.053791594	-1.286211228	437.6990
CFP 185	6.482004197	-1.925406333	642.6334	6.484775481	-1.925156364	643.5756

1 CFP 109
2 CFP 200
3 CFP 225
4 CFP 155
5 GCS 179
6 GCS 142
7 CFP 213
8 CFP 178
9 CFP 306
10 GCS 145R
11 CFP 207
12 CFP 217
13 GCS 302
14 CFP 304
15 CFP 305
16 CFP 184
17 GCS 102
18 CFP 180R
19 CFP 185

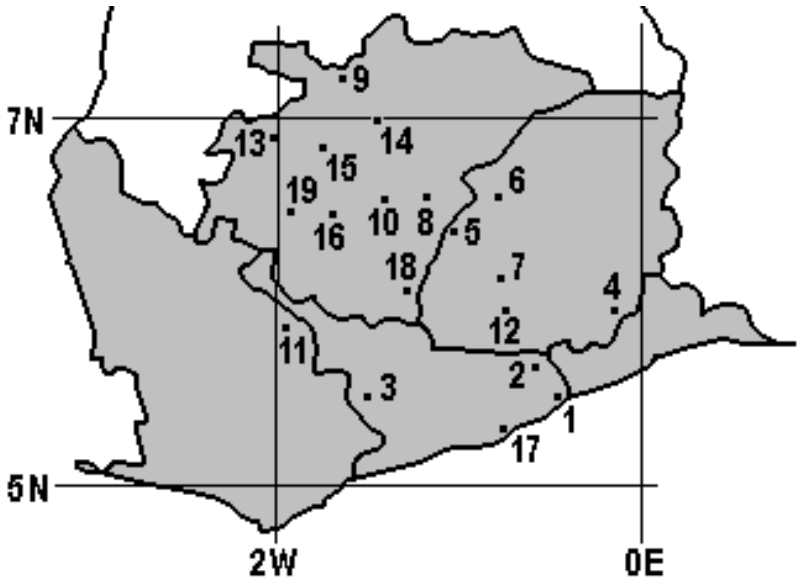


Figure C-3: Data points for Ghana's Golden Triangle, cross-referenced to official point identifications.

C.4 Sweden (SWEREF93, RT90/RH70)

This dataset consists of 20 points known in SWEREF93 (the Swedish realisation of ETRS89) and a local reference coordinate system designated RT90/RH70 by Andrei (2006). The latter is "a mixture of the Swedish triangulation network RT90 and the 2nd Swedish precise levelling network RH70". Andrei (2006) lists and applies the Cartesian coordinates as if they are derived from an ellipsoid, but it is unclear whether a geoid model had been used to convert orthometric heights to ellipsoidal heights. The ellipsoids for SWEREF93 and RT90 are

GRS80 and Bessel 1841 respectively. The distribution of the data points is shown in Figure C-4.

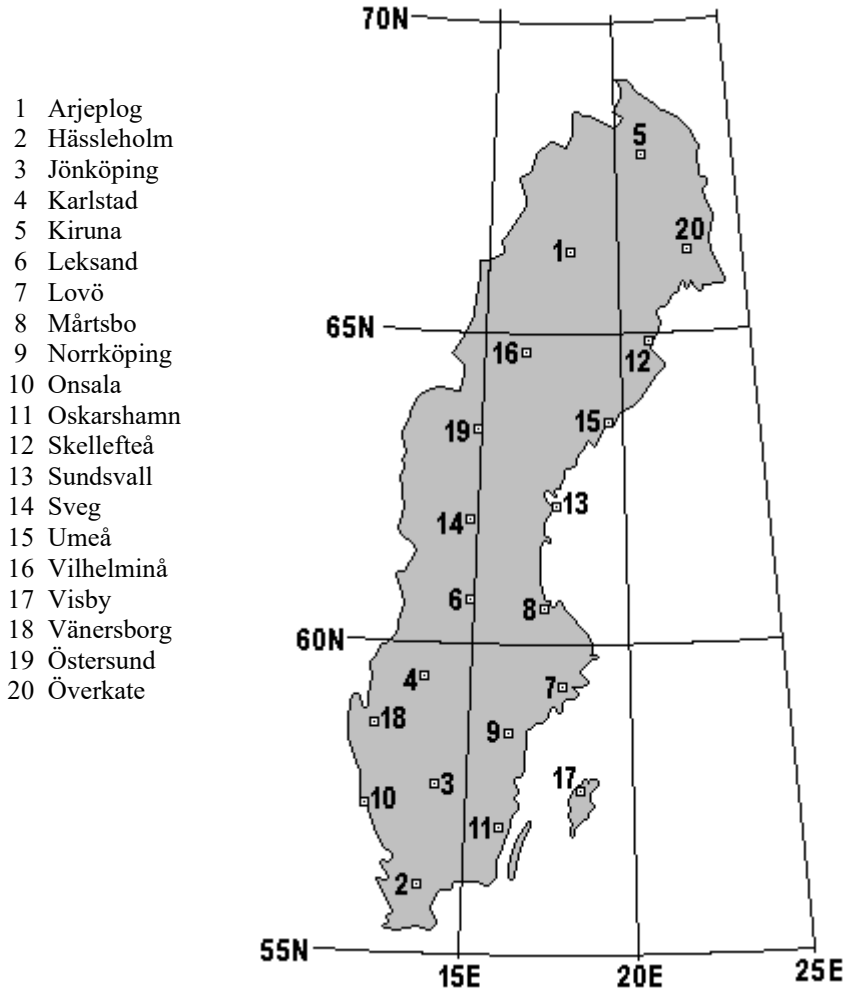


Figure C-4: Data points for Sweden, cross-referenced to the places used as identifications.

The points are the original 20 stations of the SWEPOS network as it stood at the start of 1994. (The 21st, Borås, was added in 1996.)

The geocentric Cartesian coordinates are given first, only because they constitute the original dataset given in Table 4.1 of Andrei (2006). Dr Andrei confirmed in correspondence that the columns were in the wrong order and that the following version is correct. The corrected version also appears in Amiri-Simkooei (2018).

	SWEREF93 Coordinates			RT90/RH70 Coordinates		
Point Id	X	Y	Z	X	Y	Z
1	2441775.419	799268.100	5818729.162	2441276.712	799286.666	5818162.025
2	3464655.838	845749.989	5270271.528	3464161.275	845805.461	5269712.429
3	3309991.828	828932.118	5370882.280	3309496.800	828981.942	5370322.060
4	3160763.338	759160.187	5469345.504	3160269.913	759204.574	5468784.081
5	2248123.493	865686.595	5886425.596	2247621.426	865698.413	5885856.498
6	3022573.157	802945.690	5540683.951	3022077.340	802985.055	5540121.276
7	3104219.427	998384.028	5463290.505	3103716.966	998426.412	5462727.814
8	2998189.685	931451.634	5533398.462	2997689.029	931490.201	5532835.154
9	3199093.294	932231.327	5420322.483	3198593.776	932277.179	5419760.966
10	3370658.823	711876.990	5349786.786	3370168.626	711928.884	5349227.574
11	3341340.173	957912.343	5330003.236	3340840.578	957963.383	5329442.724

12	2534031.166	975174.455	5752078.309	2533526.497	975196.347	5751510.935
13	2838909.903	903822.098	5620660.184	2838409.359	903854.897	5620095.593
14	2902495.079	761455.843	5609859.672	2902000.172	761490.908	5609296.343
15	2682407.890	950395.934	5688993.082	2681904.794	950423.098	5688426.909
16	2620258.868	779138.041	5743799.267	2619761.810	779162.964	5743233.630
17	3246470.535	1077900.355	5365277.896	3245966.134	1077947.976	5364716.214
18	3249408.275	692757.965	5426396.948	3248918.041	692805.543	5425836.841
19	2763885.496	733247.387	5682653.347	2763390.878	733277.458	5682089.111
20	2368885.005	994492.233	5818478.154	2368378.937	994508.273	5817909.286

The Cartesian coordinates were converted to geodetic coordinates by methodology described in subsection 1.4.1 using software specifically written for this study.

	SWEREF93 Coordinates			RT90/RH70 Coordinates		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)	Latitude (deg)	Longitude (deg)	Ell ht (m)
1	66.31801576	18.12486135	489.138	66.31937829	18.12871529	465.821
2	56.09221502	13.71807355	114.045	56.09234329	13.72082388	83.435
3	57.74547126	14.05960603	260.362	57.74580147	14.06243726	229.115
4	59.44401867	13.50562237	114.268	59.44456176	13.50841462	82.748
5	67.87757317	21.06023579	497.971	67.87909096	21.06478995	478.739
6	60.72214275	14.87700459	478.080	60.72283531	14.88003424	447.173
7	59.33780023	17.82891258	79.597	59.33829308	17.83232517	48.948
8	60.59514118	17.25852255	75.359	60.59579638	17.26190595	44.918
9	58.59022908	16.24637924	40.906	58.59064628	16.24953958	9.885
10	57.39529624	11.92551398	45.570	57.39559458	11.92804345	14.531
11	57.06563670	15.99680670	149.769	57.06586803	15.99988531	118.935
12	64.87919478	21.04828604	81.161	64.88033608	21.05254271	56.211
13	62.23247268	17.65988430	31.753	62.23332755	17.66340611	2.430
14	62.01741104	14.70000948	491.168	62.01826674	14.70305553	461.292
15	63.57813650	19.50959334	54.463	63.57913698	19.51349224	27.011
16	64.69784489	16.55992746	449.925	64.69902057	16.56339810	423.494
17	57.65386733	18.36731313	79.766	57.65414596	18.37073277	49.580
18	58.69312489	12.03500023	169.674	58.69358274	12.03756582	138.489
19	63.44279129	14.85806526	490.011	63.44382465	14.86118951	461.692
20	66.31785599	22.77336964	222.863	66.31914841	22.77806896	201.086

Andrei (2006) also provides the coordinates in "local topocentric coordinates" in the order N, E, U. Changing the order to E, N, U, the local level coordinates are as follows.

	System 1			System 2		
Point Id	E (or X′)	N (or Y′)	U (or Z′)	E (or X′)	N (or Y′)	U (or Z′)
1	78292.294	563600.255	-11736.010	78303.693	563599.438	-11732.331
2	-165547.693	-572086.679	-14554.597	-165559.007	-572083.961	-14558.496
3	-138070.352	-389444.783	48.624	-138078.183	-389442.388	45.902
4	-162930.880	-199314.707	8097.170	-162935.054	-199311.750	8095.194
5	196622.159	742636.961	-32692.936	196637.599	742634.029	-32687.846
6	-81954.850	-59589.289	12846.961	-81956.282	-59587.808	12846.157
7	82529.526	-213807.100	9137.567	82525.222	-213808.996	9136.609
8	48211.961	-74352.930	12632.778	48210.393	-74354.020	12632.469
9	-7691.529	-297922.030	6252.912	-7697.480	-297922.126	6251.229
10	-267482.392	-421681.043	-6335.302	-267491.043	-421676.157	-6338.615
11	-23163.971	-467361.549	-3851.951	-23173.153	-467361.472	-3854.723
12	221045.611	410450.991	-3779.987	221053.973	410446.981	-3776.884
13	66617.956	108427.764	11936.146	66620.020	108426.475	11936.876
14	-87900.945	84944.278	12494.129	-87899.491	84946.045	12494.311
15	155432.805	261424.043	5982.270	155438.029	261421.151	5984.114
16	8649.706	382401.617	2156.920	8657.257	382401.878	2159.254
17	118706.523	-400260.055	-413.138	118698.691	-400262.820	-414.835
18	-251634.874	-278118.357	2322.696	-251640.685	-278113.712	2320.308

19	-75879.036	243523.285	8566.650	-75874.384	243525.055	8567.895
20	286147.983	576529.330	-19110.901	286159.887	576524.157	-19106.586

C.5 Slovenia (D48, D96)

This dataset consists of 3331 points known in geodetic datum 1948 (D48) and geodetic datum 1996 (D96). They are based on the Bessel 1841 and GRS80 ellipsoids respectively. D48 is a local datum. D96 is a locally-adopted alternative name for ETRS89 which is a global datum. The points have an average density of one per 6 square kilometres, although actual density varies. The points form a triangulated network that is described in Berk and Komadima (2013). Their distribution was analysed computationally for this study and is illustrated in Figure C-5; numbers indicate the number of points in each 20km square.

The data was supplied by Sandi Berk of the Surveying and Mapping Authority of the Republic of Slovenia. It came in the form of Transverse Mercator grid coordinates without heights and Cartesian coordinates (based on position reduced to the ellipsoid). These were converted to geodetic coordinates by software written specifically for this research, using the algorithm in Redfearn (1948).

The set of points with their grid coordinates, Cartesian coordinates and their corresponding geodetic coordinates will be placed in the UEL data repository after completion of this thesis.

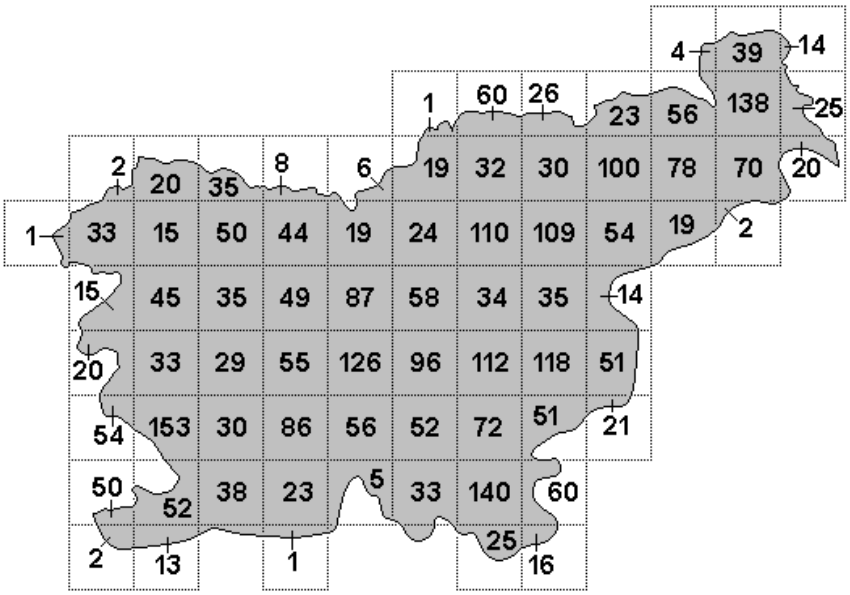


Figure C-5: Distribution of data points for Slovenia, in terms of numbers per 20km-square.

C.6 Great Britain (ETRS89, OSGB36)

This is the OSTN15 dataset, consisting of 4315 points known in European Terrestrial Reference System 1989 (ETRS89) and Ordnance Survey Great Britain 1936 (OSGB36). They are based on the GRS80 and Airy ellipsoids respectively. It was used to generate a regular grid of datum shifts at 1-km intervals which could be interpolated (Greaves *et al*, 2016). Their distribution was analysed computationally for this study and is illustrated in Figure C-6; numbers indicate the number of points in each 50km square.

Of the 4315 common points, 4269 are based on observations and form the core of a triangulated irregular network. The other 46 are "boundary points" designed to extend the network over the whole transformation grid. These points, whose coordinate shifts were generated using a 7-parameter model, are shown in Figure C-7.

The data was supplied by Ordnance Survey. It came in the form of grid coordinates (Ordnance Survey Grid Reference System 1980 and British National Grid). These were converted to geodetic coordinates by software written specifically for this research, using the algorithm in Redfearn (1948).

The set of points with both their grid coordinates and their corresponding geodetic coordinates will be placed in the UEL data repository after completion of this thesis.

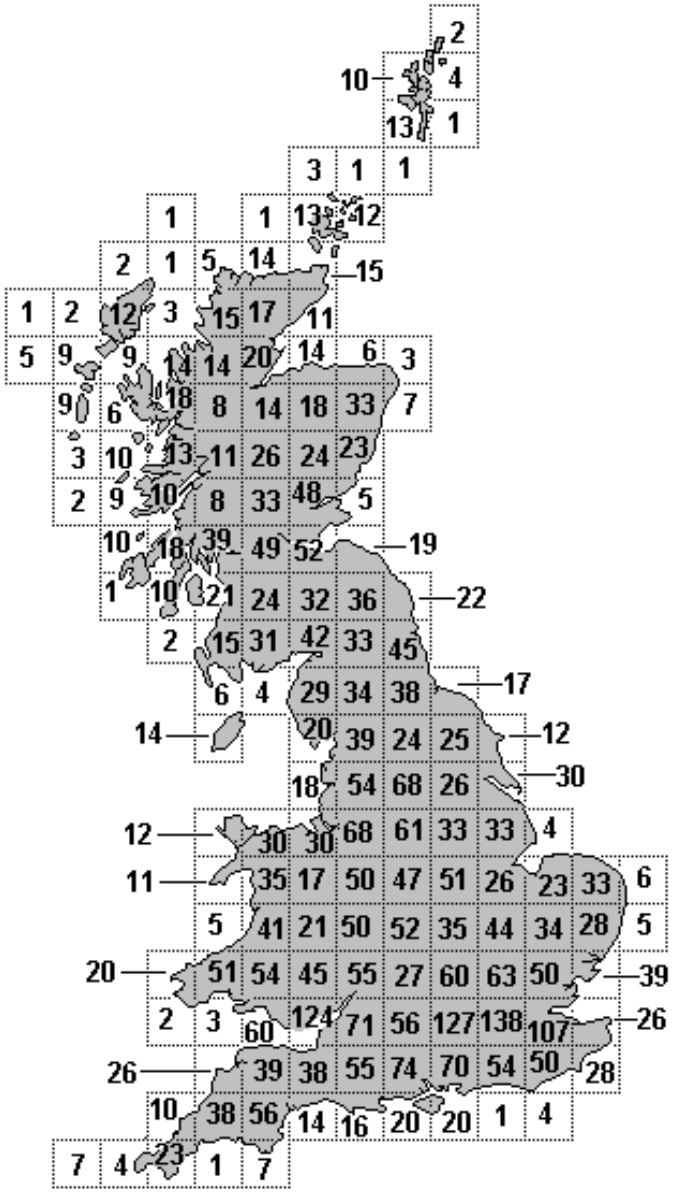


Figure C-6: Distribution of non-boundary data points for Great Britain, in terms of numbers per 50km-square.

Figure C-7: Boundary points of the Great Britain dataset, at 50km-intervals on the enclosing rectangle.

C.7 Other datasets

Other datasets were discovered during this research project. They are included here to demonstrate the different types of dataset that are used for deriving datum shifts. The sources describe a limited range of transformations and discuss their accuracy.

The following datasets are listed in ascending order of size.

C.7.1 Germany (Unknown local datum, WGS84)

The source of this 7-point dataset is Awange and Grafarend (2002). All the points in this dataset are in the southernmost quarter of Germany, on the western side. Personal correspondence with Joseph Awange failed to jog Professor Awange's memory of the local datum or the ellipsoid it was based on.

	Local System			WGS84		
Point Id	X	Y	Z	X	Y	Z
Solitude	4157222.543	664789.307	4774952.099	4157870.237	664818.678	4775416.524
Buoch Zeil	4149043.336	688836.443	4778632.188	4149691.049	688865.785	4779096.588
Hohenneuffen	4172803.511	690340.078	4758129.701	4173451.354	690369.375	4758594.075
Kuehlenberg	4177148.376	642997.635	4760764.800	4177796.064	643026.700	4761228.899
Ex Mergelaec	4137012.190	671808.029	4791128.215	4137659.549	671837.337	4791592.531
Ex Hof Asperg	4146292.729	666952.887	4783859.856	4146940.228	666982.151	4784324.099
Ex Kaisersbach	4138759.902	702670.738	4785552.196	4139407.506	702700.227	4786016.645

The following table only gives the geodetic coordinates with respect to WGS84.

	WGS84		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)
Solitude	48.78683480	9.08435741	589.286
Buoch Zeil	48.83708071	9.42538275	589.384
Hohenneuffen	48.55540861	9.39277056	821.732
Kuehlenberg	48.59248298	8.75003194	697.282
Ex Mergelaec	49.01007928	9.22270423	395.418
Ex Hof Asperg	48.91028756	9.13704010	420.106
Ex Kaisersbach	48.93117886	9.63460482	640.035

C.7.2 Sudan (Adindan, ITRF96)

The source of this 8-point dataset is Mohammed and Mohammed (2013). The Adindan datum is based on the ellipsoid Clarke 1880 Modified, also known as Clarke 1880 (RGS). The International Terrestrial Reference Frame 1996 (ITRF96) is based on the Reference Ellipsoid 1980 (GRS80).

	Adindan			ITRF96		
Point Id	X	Y	Z	X	Y	Z
1	5209207.500	3040808.244	2067652.171	5209051.179	3040794.994	2067858.390
2	5147519.189	3535228.068	1296985.925	5147351.580	3535213.493	1297189.697
3	5736045.479	2359030.970	1487557.109	5735898.786	2359026.520	1487764.835
4	4947291.159	3651035.949	1692331.703	4947124.563	3651017.896	1692537.238
5	5435162.800	3106763.337	1220913.297	5435006.649	3106749.715	1221120.126
6	5140106.403	3190427.735	2014760.964	5139946.519	3190413.669	2014966.880
7	5363495.551	3140312.750	1431563.703	5363337.027	3140299.956	1431770.658
8	5054181.363	3093217.667	2352682.277	5054033.322	3093199.747	2352881.182

The above coordinates were converted to geodetic coordinates by methodology described in subsection 1.4.1 using software specifically written for this study.

	Adindan			WGS84		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)	Latitude (deg)	Longitude (deg)	Ell ht (m)
1	19.04165997	30.27369946	373.500	19.04189579	30.27433935	381.896
2	11.81150588	34.48061319	503.550	11.81231737	34.48137359	499.493
3	13.57622538	22.35561812	1031.900	13.57690414	22.35609555	1040.083
4	15.48922654	36.42683880	571.810	15.48974444	36.42762528	574.483
5	11.10906030	29.75247805	921.011	11.10995154	29.75307903	920.412
6	18.53697532	31.82758228	354.900	18.53725174	31.82826766	361.455
7	13.05668398	30.34886251	783.701	13.05741213	30.34949912	785.255
8	21.78938338	31.46711615	290.901	21.78933496	31.46771560	302.916

There is nothing in Mohammed and Mohammed (2013) to indicate what kind of heights the Adindan Cartesian coordinates are based on. However, the height computed above for point 1 is actually the orthometric height given for the corresponding point (G021) in Ahmed (2013). It seems probable, therefore, that the Adindan local coordinates are based on orthometric heights.

C.7.3 Georgia (Pulkovo 1942, GGD)

The source of this 9-point dataset is European Union (2013). Pulkovo 1942 is based on the ellipsoid Krassovsky (1940). Georgia Geodetic Datum (GGD) is based on and aligned with the International Terrestrial Reference System, so it uses the Reference Ellipsoid 1980 (GRS80).

European Union (2013) does contain ellipsoidal heights for GGD but it replaces them by zeroes for the purpose to deriving datum transformations. The Cartesian coordinates were derived by methodology described in subsection 1.4.1 using software specifically written for this study, but they agree with those in Tables 5 and 3 of European Union (2013). Confusingly, Table 2 of that source has Cartesian coordinates that match the 3D geodetic coordinates in Table 3.

	Pulkovo 1942			GGD		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)	Latitude (deg)	Longitude (deg)	Ell ht (m)
ARMU	41.30854890	44.10268916	0.000	41.30849277	44.10147212	0.000
CHAC	41.23215780	45.97230591	0.000	41.23211588	45.97111437	0.000
FUND	41.82876705	42.85303065	0.000	41.82870615	42.85179192	0.000
GLDA	41.79006988	44.83830960	0.000	41.79002716	44.83709663	0.000
ILMA	41.42917693	45.01519911	0.000	41.42913121	45.01399441	0.000
INGU	42.72070503	42.06371530	0.000	42.72064545	42.06245402	0.000
KIZI	41.65002393	43.84159031	0.000	41.64997026	43.84036308	0.000
KODA	41.59342621	44.78251164	0.000	41.59338119	44.78130133	0.000
NORI	41.80961845	44.91722068	0.000	41.80957797	44.91600604	0.000

	Pulkovo 1942			GGD		
Point Id	X	Y	Z	X	Y	Z
G002	5056757.910	3082520.434	2361227.297	5056596.5900	3082504.8393	2361429.8197
G018	5192006.595	3041094.212	2109566.633	5191844.7290	3041078.1182	2109770.8792
G021	5209212.927	3040811.309	2067654.260	5209051.1856	3040794.9725	2067858.5289
G036	5188066.299	3158359.100	1941533.117	5187904.4037	3158343.1591	1941737.2272
G214	5162483.107	3301396.446	1765331.968	5162320.4785	3301380.8894	1765537.4975
G216	5156112.687	3307302.221	1773075.546	5155950.0611	3307286.6654	1773281.1038
G217	5155937.897	3307647.757	1772432.219	5155775.2540	3307632.2034	1772637.7713
G218	5153693.500	3315675.787	1764093.062	5153530.8379	3315660.2623	1764298.6274
G247	5011106.410	3605394.598	1600331.833	5010942.8006	3605380.1620	1600538.0991

European Union (2013) has an addendum which "contains updated parameters for transformation between GGD and Pulkovo 1942 computed from additional identical points measured since submission of the original report". However, there are no coordinates listed for the 7 additional points.

C.7.4 Sudan (Adindan, WGS84)

The source of this 13-point dataset is Ahmed (2013). Abdelrahim Elgizouli Mohamed Ahmed is also known as AbdElrahim Elgizouli Mohammed, the lead author of Mohammed and Mohammed (2013) which was cited in sub-appendix C.7.2. The Adindan datum is based on the ellipsoid Clarke 1880 Modified, also known as Clarke 1880 (RGS).

	Adindan			WGS84		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)	Latitude (deg)	Longitude (deg)	Ell ht (m)
G002	21.87236389	31.36583083	333.552	21.87237429	31.36651439	337.4061
G018	19.44294194	30.35862306	304.083	19.44314893	30.35926984	306.2209
G021	19.04165944	30.27369861	380.073	19.04189699	30.27433915	381.9369
G036	17.84054722	31.33196194	358.491	17.84087116	31.33262765	360.9299
G214	16.17517361	32.59882444	480.614	16.17563776	32.59952114	483.3360
G216	16.24786500	32.67746306	549.834	16.24832330	32.67816186	552.7296
G217	16.24218389	32.68106611	407.737	16.24264263	32.68176510	410.6173
G218	16.16363806	32.75557306	434.357	16.16410295	32.75627386	437.2637
G247	14.62833250	35.73428889	517.333	14.62891902	35.73506698	522.6840
G249	14.38838722	35.94811556	567.736	14.38898611	35.94890406	573.1138
G652	13.24267222	33.09916667	500.504	13.24336855	33.09988708	500.6076
G901	12.57951417	34.09889167	447.070	12.58024448	34.09962601	447.5745
G905	12.81700833	33.98224972	438.107	12.81770863	33.98297823	438.7897

	Adindan			WGS84		
Point Id	X	Y	Z	X	Y	Z
G002	5056757.910	3082520.434	2361227.297	5056596.5900	3082504.8393	2361429.8197
G018	5192006.595	3041094.212	2109566.633	5191844.7290	3041078.1182	2109770.8792
G021	5209212.927	3040811.309	2067654.260	5209051.1856	3040794.9725	2067858.5289
G036	5188066.299	3158359.100	1941533.117	5187904.4037	3158343.1591	1941737.2272
G214	5162483.107	3301396.446	1765331.968	5162320.4785	3301380.8894	1765537.4975
G216	5156112.687	3307302.221	1773075.546	5155950.0611	3307286.6654	1773281.1038
G217	5155937.897	3307647.757	1772432.219	5155775.2540	3307632.2034	1772637.7713
G218	5153693.500	3315675.787	1764093.062	5153530.8379	3315660.2623	1764298.6274
G247	5011106.410	3605394.598	1600331.833	5010942.8006	3605380.1620	1600538.0991
G249	5003040.086	3628003.915	1574642.485	5002876.2239	3627990.1446	1574848.0210
G652	5202482.046	3391347.557	1451536.300	5202318.3388	3391334.0501	1451741.1175
G901	5156050.122	3490762.917	1380015.242	5155887.2204	3490749.0003	1380217.9958
G905	5158349.719	3477026.187	1405643.756	5158187.2645	3477012.0684	1405845.4032

The Adindan coordinates are new ellipsoidal coordinates, resulting from an iterative process described in Ahmed (2013) to generate ellipsoidal heights.

From the Cartesian coordinates, it is apparent that G021 is also the first point in the 8-point Sudan dataset in sub-appendix C.7.2. This is the only point common to both datasets, although in the smaller dataset the orthometric height is used.

C.7.5 Brunei (BT48, GDBD2009)

The source of this 18-point dataset is Brunei Darussalam Survey Department (2009). The datum BT48 is based on the Everest (Borneo) ellipsoid, sometimes called Modified Everest (Brunei). Geocentric Datum Brunei Darussalam 2009 (GDBD2009) is based on the Reference Ellipsoid 1980 (GRS80).

	BT48			GDBD2009		
Point Id	Latitude (deg)	Longitude (deg)	Ell ht (m)	Latitude (deg)	Longitude (deg)	Ell ht (m)
B001	4.90384417	114.79154250	112.314	4.90300447	114.79458016	159.552
B004	4.13759569	114.33505783	15.687	4.13680964	114.33815845	61.436
B007	4.01299117	114.49381228	290.169	4.01221614	114.49689118	336.963
B008	4.49301308	114.81589944	526.943	4.49220444	114.81893377	574.667
B010	4.78255408	115.34335044	721.473	4.78172082	115.34631303	772.482
B011	4.90363522	115.51269836	880.298	4.90279052	115.51563754	931.992
B012	4.01913858	114.28624872	92.647	4.01836164	114.28935546	138.429
B019	4.19972242	113.99408986	463.241	4.19893132	113.99723613	506.891
B021	4.39140792	113.99498025	81.748	4.39060496	113.99812713	124.704
B025	4.85479678	114.93067681	215.082	4.85396071	114.93369545	263.166
B026	4.80363889	115.23563611	52.870	4.80280355	115.23861399	103.106

B027	4.60547361	115.02840806	531.581	4.60465531	115.03141396	580.949
B032	4.69377417	114.52176389	79.904	4.69295230	114.52483945	125.499
B033	4.37535389	114.46813972	254.701	4.37455159	114.47121949	300.629
B034	5.01092578	115.04624436	142.371	5.01007712	115.04924840	190.806
B036	5.28431897	115.18233578	66.265	5.28344591	115.18532304	114.739
B038	4.32778208	114.64112392	49.443	4.32698405	114.64418113	96.369
B039	4.27873936	114.82527186	373.864	4.27792472	114.82833308	422.082

Brunei Darussalam Survey Department (2009) explains that the orthometric heights for the local datum were converted to ellipsoidal heights by the use of a DMA-developed geoid height model.

C.7.6 Italy (Genova 1902, WGS84)

The source of this 30-point dataset is Timár *et al* (2011). The points are well spread over Italy (including Sicily) but contain no points in Sardinia. The paper actually lists 31 points, but the authors deduced an error in the coordinates of point 23 and excluded it from computation of transformations. The Genova 1902 datum is based on the Bessel 1841 ellipsoid.

		Genova 1902		WGS84	
		Latitude (deg)	Longitude (deg)	Latitude (deg)	Longitude (deg)
1	Vercelli PI	45.450000	8.204363	45.450433	8.205047
2	Pordenone	45.954150	12.660401	45.954554	12.660503
3	Monte Bronzone	45.708633	9.990309	45.709048	9.990771
4	Lodi	45.313672	9.502465	45.314130	9.502994
5	Alessandria	44.914226	8.610709	44.914720	8.611350
6	Monte Bignone	43.872907	7.732943	43.873513	7.733709
7	Forte Diamante	44.460561	8.938858	44.461114	8.939474
8	Portonovo	44.531880	11.753116	44.532469	11.753367
9	Siena (T. del Mangia)	43.317535	11.331871	43.318285	11.332211
10	Urbino	43.724425	12.636111	43.725135	12.636264
11	Monte Pennino	43.100577	12.888616	43.101383	12.888754
12	Roma (Monte Mario)	41.923444	12.451900	41.924405	12.452135
13	Monte Ocre	42.255529	13.443024	42.256464	13.443123
14	Valle Palombo	41.650375	14.259624	41.651030	14.259638
15	Monte Terminio	40.840440	14.937349	40.841579	14.937321
16	Taranto	40.474949	17.228763	40.476222	17.228541
17	Lecce	40.350682	18.169618	40.352009	18.169302
18	Monte Brutto	39.139509	16.421972	39.140896	16.421881
19	Torre Titone	37.847500	12.539363	37.849035	12.539843
20	Monte Etna (P. Lucia)	37.763122	14.985233	37.764674	14.985380
21	Monte Castelluccio	37.414518	13.779165	37.416115	13.779475
22	Mineo	37.265472	14.692406	37.267086	14.692592
24†	Innsbruck	47.270276	11.393789	47.270495	11.394035
25†	Krimberg	45.929041	14.471915	45.929449	14.471782
26†	Monte Cairo	41.540496	13.760462	41.541532	13.760545
27†	Francolise	41.181580	14.063770	41.182662	14.063830
28†	Cancello	40.992680	14.430113	40.993799	14.430148
29†	Miradois, Napoli	40.862511	14.255439	40.863634	14.255497
30†	Monte Petrella	41.321142	13.665475	41.322197	13.665572
31†	Marigliano	40.924113	14.456006	40.925236	14.456036

† Point number used in Timár *et al* (2011); point 23 (Sardinia Punte Ideale) was excluded from transformation derivation due to an "obvious blunder" in its coordinates (taken from http://www.fiduciali.it).

C.7.7 Ghana (Leigon, Accra)

The source of this dataset is Kumi-Boateng and Ziggah (2020). The Leigon datum is based on the ellipsoid Clarke 1880 Modified, also known as Clarke 1880 (RGS). Accra is based on

the ellipsoid War Office 1924. The coordinates of the data points are derived from Transverse Mercator projections with the same central meridian (1°W).

Kumi-Boateng and Ziggah (2020) lists 46 points and assigns identifications R1-R31 and T1-T15. This was done to differentiate the 31 points chosen as "training" points (used as control points to derive transformations) from the 15 "testing" points (used to provide an independent check on accuracy).

	Leigon TM grid coordinates		Accra TM grid coordinates	
Point ID	Eastings (m)	Northings (m)	Eastings (m)	Northings (m)
R1	96308.47472	343341.49567	96315.96670	343342.49846
R2	102396.73609	348793.77721	102404.08481	348794.59407
R3	98384.46857	357002.14304	98391.82949	357002.79531
R4	108527.33719	363356.54637	108534.39940	363357.18949
R5	179593.84552	181252.23939	179598.74366	181251.11773
R6	105410.76018	324978.11495	105418.16987	324979.30977
R7	145906.45533	312862.32965	145913.93512	312863.29282
R8	217305.38737	141614.96995	217310.64517	141612.59251
R9	105737.94133	323098.99428	105745.36321	323100.32320
R10	161720.22307	392995.67524	161727.17860	392996.58355
R11	161317.32358	385896.81692	161324.16329	385897.71913
R12	196537.92640	337894.67918	196545.33608	337896.21233
R13	164804.79001	193489.59406	164809.62414	193488.75586
R14	153010.64460	405832.84525	153017.16732	405833.79013
R15	149300.93347	196924.39110	149305.45974	196923.63519
R16	181138.36622	153293.64504	181143.18815	153291.73394
R17	211446.81392	172735.17242	211452.26069	172733.26437
R18	210262.37140	256084.33475	210269.60125	256083.42035
R19	156103.67278	390525.41992	156110.53992	390526.21240
R20	140426.63819	371486.73412	140433.33159	371487.39859
R21	66995.11272	166458.22784	66997.02687	166456.60630
R22	163685.79714	196883.82530	163690.61907	196883.06635
R23	186272.81482	226968.02576	186278.87729	226967.29119
R24	153046.17206	242613.60198	153052.11870	242614.12928
R25	199279.75217	197449.17694	199285.24161	197447.96689
R26	137145.43581	391777.91507	137151.97072	391778.46371
R27	118155.82277	176791.46216	118159.51389	176790.37402
R28	80820.82438	306190.56582	80828.61201	306191.85512
R29	210755.75200	300209.21400	210763.86141	300210.03286
R30	283778.08500	204680.01600	283785.99819	204677.95919
R31	240727.35400	192138.14000	240734.20698	192136.52110
T1	82660.60151	329013.92825	82668.25503	329015.13830
T2	198067.01204	296216.39775	198074.93379	296216.90067
T3	175114.74373	237505.14060	175120.91287	237505.09488
T4	188231.00359	270605.93157	188238.04446	270605.90413
T5	200781.58871	171978.38218	200786.91661	171976.62043
T6	224664.95918	206357.89630	224671.11003	206356.31439
T7	111533.39991	187113.71454	111536.93559	187113.00131
T8	143581.25419	216569.24427	143585.68903	216569.09187
T9	124931.98050	341128.18040	124939.24083	341129.32340
T10	130558.08321	167316.40970	130562.14315	167315.11735
T11	149600.00294	261798.37828	149606.27267	261799.17380
T12	161269.90589	302076.12944	161276.89800	302076.89754
T13	179379.85784	347155.69824	179387.14865	347157.03631
T14	161035.63836	379092.87365	161042.53750	379093.78195
T15	167630.72830	389866.04367	167637.56801	389866.95502

APPENDIX D: SIMULATED DATASETS

This appendix contains simulated datasets of control points in separate datums in arbitrarily-chosen regions with no relationship to political boundaries. Each local datum will either be identified by datum or by ellipsoid, so that the ellipsoid corresponds to one that is used in that part of the world.

The primary purpose of these datasets was to test the new method introduced in Section 5.1 to derive Helmert transformations. For that reason, they were generated with rotations and scale changes that were large compared with most actual cases. Measurement errors were simulated by the addition of pseudo-random numbers to the target coordinates.

D.1 Helmatan

Helmatan is a fictitious province in Western Asia. The source datum uses the Krassovsky ellipsoid. The target datum is WGS84. 12 points were selected in the area bounded by latitudes 50°N to 42°N and longitudes 71°E to 84°E. The distribution of the data points is shown in Figure D-1.

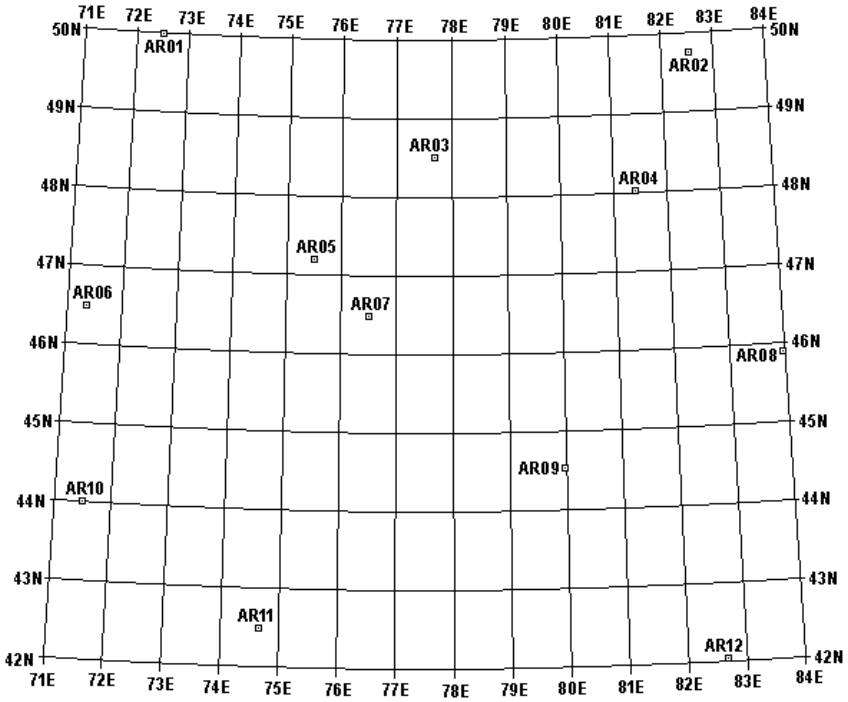


Figure D-1: Data points for Helmatan.

The geodetic coordinates are as follows.

	Krassovsky geodetic coordinates			WGS84 geodetic coordinates for Simulation 1		
	ϕ (source) (degrees)	λ (source) (degrees)	h (source) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
AR01	49.98530501	72.49755439	107.471	49.986081074	72.496426098	20.758
AR02	49.75037448	82.52817495	88.803	49.750862312	82.527301759	8.246
AR03	48.51786466	77.66803433	129.106	48.518470151	77.667033334	50.248
AR04	48.01245772	81.39976008	150.328	48.012938638	81.398860247	74.520
AR05	47.19973218	75.48894298	106.973	47.200407136	75.487691328	28.301
AR06	46.49607722	71.32990421	139.518	46.496788995	71.328471073	61.504
AR07	46.40268438	76.50234447	102.097	46.403294102	76.501141060	29.867
AR08	45.89718599	83.98245377	114.006	45.897541697	83.981374805	46.043
AR09	44.51366007	79.98544078	98.869	44.514150054	79.984143300	30.105
AR10	44.00741883	71.48894376	143.803	44.008133016	71.487484713	77.405

AR11	42.49664008	74.66588039	127.058	42.497203752	74.664379188	67.421
AR12	42.03583291	82.66854337	133.908	42.036177454	82.667183883	78.120

	WGS84 geodetic coordinates for Simulation 2			WGS84 geodetic coordinates for Simulation 3		
	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
AR01	49.9893019952	72.4956714115	204.061	49.9915287256	72.4909653098	233.837
AR02	49.7537369824	82.5271598842	191.427	49.7557603321	82.5228826395	221.131
AR03	48.5215252341	77.6664958975	233.516	48.5236559567	77.6619530087	263.265
AR04	48.0158580373	81.3984988005	257.745	48.0179083337	81.3940832848	287.468
AR05	47.2035366178	75.4869262659	211.614	47.2057109490	75.4822359667	241.386
AR06	46.5000465833	71.3274138679	244.874	46.5022944901	71.3225269337	274.674
AR07	46.4063905389	76.5003783012	213.177	46.4085456839	76.4956958716	242.943
AR08	45.9003610239	83.9809740274	229.258	45.9023513519	83.9765489886	258.965
AR09	44.5171249205	79.9834327135	213.391	44.5192085428	79.9788011536	243.135
AR10	44.0113879334	71.4862932013	260.801	44.0136345701	71.4813313847	290.605
AR11	42.5003633076	74.6632682784	250.792	42.5025554691	74.6583723603	280.575
AR12	42.0390517834	82.6664247321	261.389	42.0410754911	82.6617774560	291.111

D.2 Helmatia

Helmatia is a fictitious region in South America. The source datum uses the South American 1969 ellipsoid. The target datum is WGS84. 16 points were selected in the area bounded by latitudes 0°N to 10°N and longitudes 75°W to 65°W. The distribution of the data points is shown in Figure D-2. It is deliberately uneven, in contrast to Helmatan.

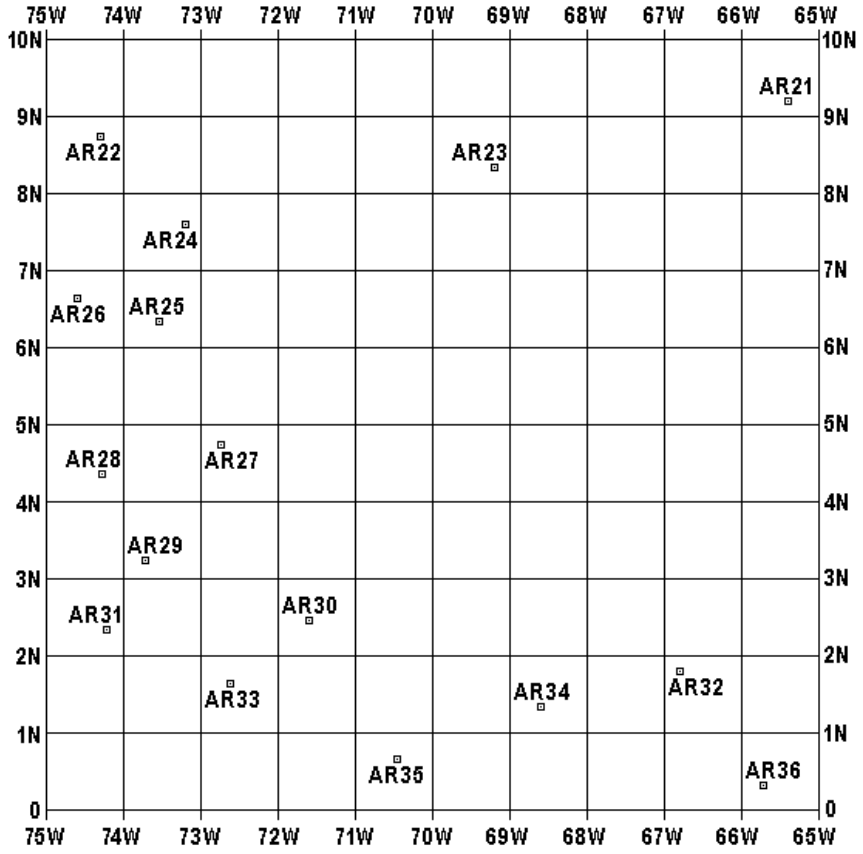


Figure D-2: Data points for Helmatia.

The geodetic coordinates are as follows.

	SAD69 geodetic coordinates			WGS84 geodetic coordinates for Simulation 1		
	ϕ (source) (degrees)	λ (source) (degrees)	h (source) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
AR21	9.20452136	-65.40652096	103.211	9.20377415	-65.40393112	138.005
AR22	8.74991840	-74.29653093	131.008	8.74860146	-74.29368906	180.407
AR23	8.33167287	-69.19741048	112.341	8.33065962	-69.19464310	157.065
AR24	7.59750185	-73.20542803	99.327	7.59626607	-73.20254236	151.616
AR25	6.63168834	-74.59873070	87.963	6.63043382	-74.59579712	132.393
AR26	6.33885628	-73.54073874	105.881	6.33764767	-73.53779632	153.391
AR27	4.75168461	-72.74863900	129.230	4.75058689	-72.74563836	170.648
AR28	4.35994820	-74.27147905	114.947	4.35874703	-74.26839902	163.315
AR29	3.23175580	-73.72954021	106.018	3.23059424	-73.72643743	145.296
AR30	2.46005732	-71.70359943	140.896	2.45902420	-71.70051549	180.565
AR31	2.34168447	-74.22880612	139.275	2.34057373	-74.22562971	180.867
AR32	1.79635190	-66.80581963	136.060	1.79556884	-66.80274424	162.045
AR33	1.63246410	-72.61798439	147.903	1.63142578	-72.61483275	185.603
AR34	1.33895611	-68.59620277	142.075	1.33810125	-68.59309329	176.645
AR35	0.66930407	-70.45260680	135.869	0.66833790	-70.44940210	168.707
AR36	0.32085077	-65.71906204	96.153	0.32021073	-65.71591348	123.607

	WGS84 geodetic coordinates for Simulation 2			WGS84 geodetic coordinates for Simulation 3		
	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
AR21	9.201705053	-65.401962259	128.189	9.201249213	-65.397351798	118.560
AR22	8.746060078	-74.291639923	170.551	8.745265639	-74.286996803	160.887
AR23	8.328382251	-69.192603935	147.250	8.327780174	-69.187954134	137.611
AR24	7.593779386	-73.200441624	141.801	7.593025571	-73.195757313	132.153
AR25	6.627877048	-74.593640238	122.602	6.627071270	-74.588917517	112.957
AR26	6.335143828	-73.535630033	143.614	6.334377415	-73.530898053	133.975
AR27	4.748123076	-72.743395092	160.923	4.747386254	-72.738604916	151.301
AR28	4.356206277	-74.266130879	153.598	4.355412556	-74.261324574	143.977
AR29	3.228080559	-73.724114807	135.616	3.227306985	-73.719267343	126.007
AR30	2.456613995	-71.698158135	170.913	2.455916452	-71.693283305	161.314
AR31	2.338034960	-74.223261890	171.215	2.337242767	-74.218381309	161.614
AR32	1.793421061	-66.800358925	152.420	1.792910775	-66.795460941	142.831
AR33	1.628968423	-72.612431906	175.976	1.628236425	-72.607525876	166.383
AR34	1.335855594	-68.590681437	167.031	1.335276341	-68.585765598	157.443
AR35	0.665993069	-70.446953651	159.111	0.665342900	-70.442012217	149.529
AR36	0.318123354	-65.713447919	114.023	0.317655160	-65.708493684	104.444

D.3 Helmatto

Helmatto is a fictitious region in North Africa. The source datum is Arc 1950 which uses the Clarke 1880 (Arc) ellipsoid. The target datum is WGS84. 14 points were selected in the area Helmatto is a fictitious region in North Africa. The source datum is Arc 1950 which uses the bounded by latitudes 0°N to 3°N and longitudes 15°E to 27°E. The distribution of the data points is shown in Figure D-3.

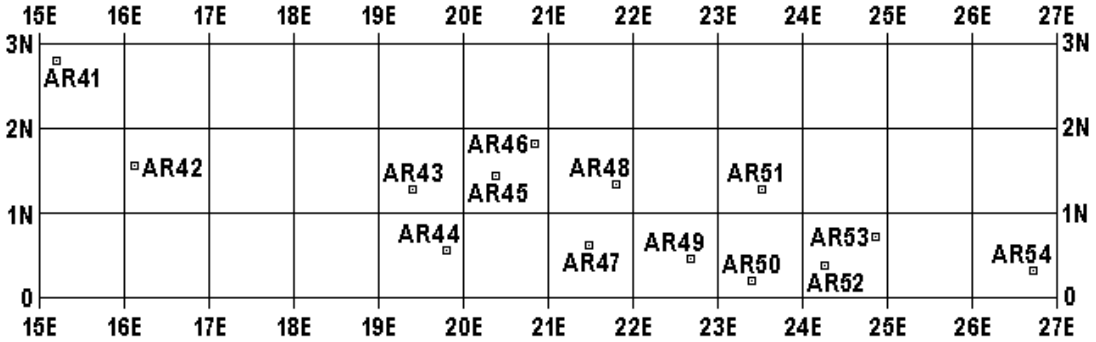


Figure D-3: Data points for Helmatto.

The geodetic coordinates are as follows.

	Arc 1950 geodetic coordinates			WGS84 geodetic coordinates for Simulation 1		
	ϕ (source) (degrees)	λ (source) (degrees)	h (source) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
AR41	2.79142941	15.20818396	93.211	2.79482055	15.20344720	21.314
AR42	1.56386680	16.11201291	121.008	1.56746833	16.10734578	49.991
AR43	1.28797460	19.40449487	103.341	1.29166886	19.39986586	22.762
AR44	0.56972872	19.79402716	91.327	0.57357401	19.78943872	14.727
AR45	1.44713719	20.38700712	80.963	1.45081318	20.38243316	2.952
AR46	1.82219197	20.83703740	97.881	1.82585170	20.83241945	19.551
AR47	0.62729575	21.47612070	120.230	0.63117386	21.47159347	45.188
AR48	1.34085656	21.80624898	104.947	1.34465055	21.80170774	32.567
AR49	0.45926101	22.68583028	197.518	0.46312043	22.68131127	117.518
AR50	0.20051714	23.40206190	131.596	0.20447992	23.39754354	49.021
AR51	1.27222226	23.52734654	130.875	1.27608109	23.52282880	55.993
AR52	0.38637905	24.26407616	128.060	0.39030307	24.25958197	41.937
AR53	0.71138484	24.85025879	137.503	0.71533576	24.84579248	50.485
AR54	0.31884231	26.72708640	133.375	0.32287991	26.72265435	44.230

	WGS84 geodetic coordinates for Simulation 2			WGS84 geodetic coordinates for Simulation 3		
	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
AR41	2.797566352	15.201058822	11.216	2.799616116	15.196486390	1.096
AR42	1.570253640	16.105011976	39.851	1.572315890	16.100457474	29.698
AR43	1.294591967	19.397547487	12.614	1.296695222	19.392999320	2.455
AR44	0.576512799	19.787149182	4.552	0.578620451	19.782609243	-5.627
AR45	1.453775552	20.380109697	-7.189	1.455889698	20.375560645	-17.345
AR46	1.828831752	20.830082087	9.424	1.830950673	20.825529568	-0.721
AR47	0.634178752	21.469302609	35.016	0.636304254	21.464762715	24.838
AR48	1.347668099	21.799390144	22.422	1.349796883	21.794843444	12.264
AR49	0.466171273	22.679027299	107.339	0.468308482	22.674489431	97.157
AR50	0.207557323	23.395269258	38.832	0.209701015	23.390733854	28.643
AR51	1.279163082	23.520515801	45.847	1.281307866	23.515971181	35.687
AR52	0.393411806	24.257301255	31.755	0.395562857	24.252764451	21.571
AR53	0.718465404	24.843500640	40.317	0.720621180	24.838961460	30.142
AR54	0.326074274	26.720376787	34.046	0.328243659	26.715841051	23.860

D.4 Helmatrun

Helmatrun is a region that coincides geographically with Réunion Island, but the 13 "data points" are fictitious. (They were chosen to ensure that the minimum convex polynomial was never too far from the coastline). The source datum is Réunion 1947 which uses the International ellipsoid. The target datum is WGS84. The distribution of the data points is shown in Figure D-4.

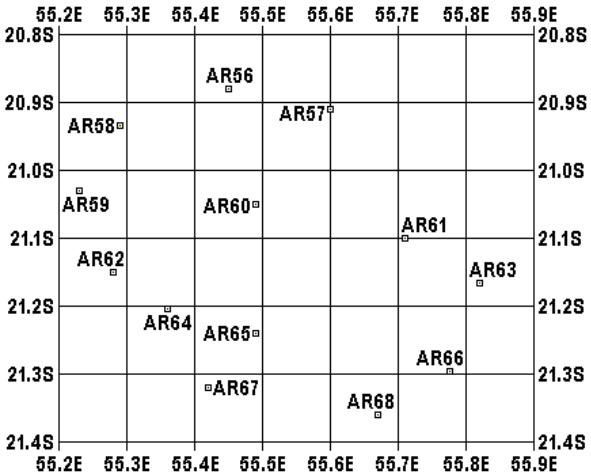


Figure D-4: Data points for Helmatrun.

The geodetic coordinates are as follows.

	Réunion geodetic coordinates			WGS84 geodetic coordinates for Simulation 1		
	ϕ (source) (degrees)	λ (source) (degrees)	h (source) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
AR56	-20.88086508	55.45048687	17.328	-20.893320376	55.444576176	18.284
AR57	-20.90976380	55.59925408	24.860	-20.922183500	55.593366339	23.802
AR58	-20.93443729	55.29046271	15.605	-20.946936737	55.284551034	18.867
AR59	-20.02949407	55.23057307	16.438	-20.042016591	55.224395096	20.517
AR60	-20.05086849	55.49065088	46.831	-20.063326568	55.484504743	46.442
AR61	-21.10077436	55.71064009	22.506	-21.113164836	55.704815324	18.984
AR62	-21.14936055	55.27961168	16.537	-21.161853053	55.273763672	20.196
AR63	-21.16974035	55.82083780	18.537	-21.182093110	55.815039501	13.455
AR64	-21.20416749	55.35943802	14.705	-21.216637370	55.353612518	16.995
AR65	-21.23935082	55.49073218	26.518	-21.251790879	55.484928098	26.787
AR66	-21.29605386	55.75964082	15.762	-21.308420152	55.753870215	11.866
AR67	-21.31976036	55.41974093	17.338	-21.332215161	55.413947785	19.043
AR68	-21.36040794	55.67037679	16.066	-21.372797664	55.664618813	13.555

	WGS84 geodetic coordinates for Simulation 2			WGS84 geodetic coordinates for Simulation 3		
	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
AR56	-20.888630085	55.443143220	39.548	-20.893320376	55.444576176	18.284
AR57	-20.917511237	55.591924837	45.070	-20.922183500	55.593366339	23.802
AR58	-20.942227152	55.283115701	40.124	-20.946936737	55.284551034	18.867
AR59	-20.037299120	55.223084571	41.813	-20.042016591	55.224395096	20.517
AR60	-20.058640534	55.483183635	67.744	-20.063326568	55.484504743	46.442
AR61	-21.108506215	55.703344118	40.246	-21.113164836	55.704815324	18.984
AR62	-21.157142318	55.272299243	41.443	-21.161853053	55.273763672	20.196
AR63	-21.177447916	55.813555367	34.718	-21.182093110	55.815039501	13.455
AR64	-21.211936314	55.352138057	38.242	-21.216637370	55.353612518	16.995
AR65	-21.247105721	55.483444671	48.037	-21.251790879	55.484928098	26.787
AR66	-21.303767622	55.752370541	33.122	-21.308420152	55.753870215	11.866
AR67	-21.327521478	55.412455524	40.287	-21.332215161	55.413947785	19.043
AR68	-21.368134354	55.663113045	34.805	-21.372797664	55.664618813	13.555

D.5 St Fuitioci

St Fuitioci is a fictitious island covering a much smaller area than any other region in this appendix. The source datum uses the International ellipsoid. The target datum is WGS84. The distribution of the 12 data points is shown in Figure D-5.

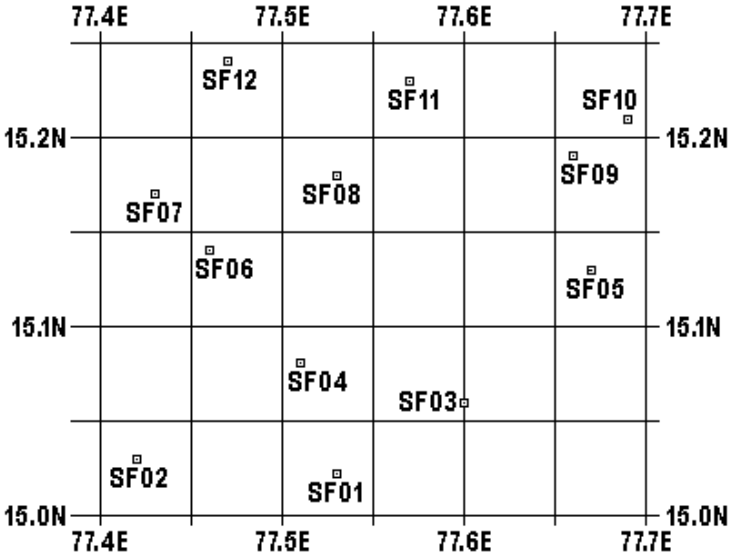


Figure D-5: Data points for St Fuitioci.

The geodetic coordinates are as follows.

	International Ellipsoid geodetic coordinates			WGS84 geodetic coordinates for Simulation 1		
	ϕ (source) (degrees)	λ (source) (degrees)	h (source) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
SF01	15.02154897	77.53030768	27.328	15.05045987	77.53233657	105.911
SF02	15.02993272	77.42026583	24.125	15.05888479	77.42224922	104.684
SF03	15.05984307	77.60037418	13.605	15.08879002	77.60250007	87.584
SF04	15.08080416	77.51018793	16.438	15.10972668	77.51221864	96.667
SF05	15.12989437	77.66983764	46.831	15.15879699	77.67193636	123.840
SF06	15.14035727	77.46020718	25.506	15.16935112	77.46224141	107.840
SF07	15.16987420	77.42993710	16.537	15.19882171	77.43200025	99.062
SF08	15.18038739	77.52987428	20.537	15.20934932	77.53195028	106.341
SF09	15.18983658	77.66026962	14.705	15.21875649	77.66245405	88.806
SF10	15.20987405	77.68979635	26.518	15.23877683	77.69193244	105.573
SF11	15.23006493	77.57020782	15.264	15.25902672	77.57235722	98.604
SF12	15.23985793	77.46990517	18.338	15.26883915	77.47205572	99.474

	WGS84 geodetic coordinates for Simulation 2			WGS84 geodetic coordinates for Simulation 3		
	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
SF01	15.03593044	77.53099830	37.603	15.03328670	77.52946450	32.317
SF02	15.04433571	77.42092891	35.508	15.04168568	77.41939952	30.126
SF03	15.07426838	77.60114248	19.208	15.07162725	77.59960334	13.940
SF04	15.09518741	77.51087324	27.383	15.09254068	77.50933689	22.024
SF05	15.14427970	77.67055376	54.936	15.14164014	77.66900746	49.655
SF06	15.15479694	77.46089411	37.369	15.15214558	77.45935655	31.910
SF07	15.18425920	77.43065300	27.968	15.18160524	77.42911508	22.455
SF08	15.19480270	77.53058340	35.775	15.19215387	77.52904041	30.333
SF09	15.20423107	77.66106231	18.990	15.20158907	77.65951295	13.642
SF10	15.22425432	77.69053183	35.670	15.22161330	77.68897999	30.326
SF11	15.24448162	77.57097420	27.606	15.24183341	77.56942653	22.147
SF12	15.25427592	77.47068875	27.656	15.25162191	77.46914498	22.106

D.6 Main Gyria

Main Gyria is a fictitious region enclosed by a degree square near the Equator. The source datum uses the Bessel 1841 ellipsoid. The target datum is WGS84. The distribution of the 20 data points is shown in Figure D-6.

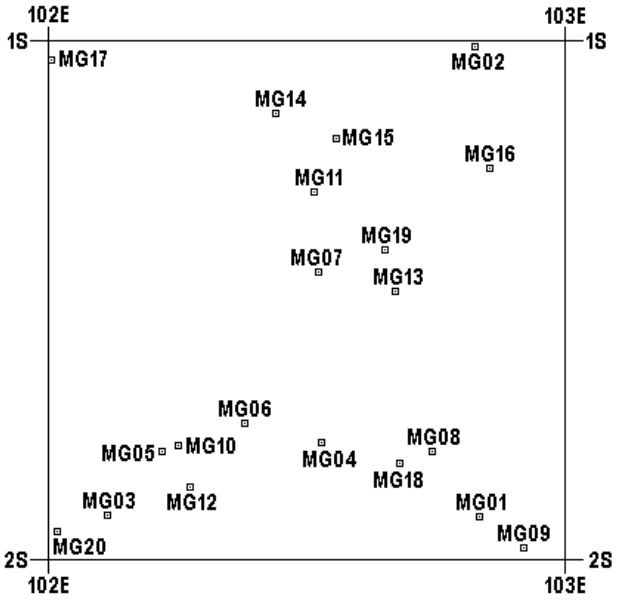


Figure D-6: Data points for Main Gyria.

The geodetic coordinates are as follows.

	Bessel 1941 Ellipsoid geodetic coordinates			WGS84 geodetic coordinates for Simulation 1		
	ϕ (source) (degrees)	λ (source) (degrees)	h (source) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
MG01	-1.91840762	102.83554804	43.917	-1.91969280	102.84250036	99.443
MG02	-1.01049447	102.82486671	45.839	-1.01182429	102.83214335	99.661
MG03	-1.91436249	102.11326265	53.928	-1.91539782	102.12020191	109.021
MG04	-1.77550519	102.52983457	23.010	-1.77663523	102.53682504	76.604
MG05	-1.79023165	102.22007596	34.702	-1.79131375	102.22706994	87.547
MG06	-1.73682499	102.38029593	49.927	-1.73799047	102.38733165	103.429
MG07	-1.44438392	102.52319837	8.270	-1.44560237	102.53033719	60.991
MG08	-1.79282272	102.74346119	44.558	-1.79407508	102.75045360	93.534
MG09	-1.97845048	102.92099869	28.536	-1.97972676	102.92790709	77.669
MG10	-1.77955031	102.25211996	28.934	-1.78060768	102.25910448	83.383
MG11	-1.29080015	102.51510811	41.596	-1.29203873	102.52228056	94.252
MG12	-1.86095583	102.27348262	25.089	-1.86207501	102.28039909	79.270
MG13	-1.48306412	102.67273700	41.354	-1.48433754	102.67983059	94.474
MG14	-1.13867044	102.44033879	55.055	-1.13985918	102.44750972	100.539
MG15	-1.18803197	102.55783343	33.906	-1.18923928	102.56503320	79.587
MG16	-1.24548376	102.85431963	8.669	-1.24677050	102.86150141	61.522
MG17	-1.03849334	102.00644934	13.155	-1.03956143	102.01362287	65.430
MG18	-1.81418538	102.67937320	56.094	-1.81540073	102.68635364	107.700
MG19	-1.40165859	102.65137434	45.199	-1.40292548	102.65848867	91.542
MG20	-1.94640648	102.01713067	11.232	-1.94742951	102.02406770	67.350

	WGS84 geodetic coordinates for Simulation 2			WGS84 geodetic coordinates for Simulation 3		
	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
MG01	-1.918499729	102.840089330	42.236	-1.918021336	102.838344383	74.055
MG02	-1.010678593	102.829638312	42.610	-1.010232554	102.827846029	74.533
MG03	-1.914281179	102.117827833	52.724	-1.913841365	102.116108164	85.168
MG04	-1.775481689	102.534414922	19.818	-1.775024563	102.532673245	51.925
MG05	-1.790192159	102.224677378	31.144	-1.789751011	102.222947405	63.514
MG06	-1.736854681	102.384925232	46.836	-1.736406873	102.383186797	79.076
MG07	-1.444466406	102.527892901	4.271	-1.444021239	102.526134046	36.421
MG08	-1.792898113	102.748034251	36.470	-1.792429013	102.746285953	68.387
MG09	-1.978521635	102.925497887	20.339	-1.978036598	102.923753078	52.074
MG10	-1.779483251	102.256709139	26.943	-1.779040772	102.254977473	59.288
MG11	-1.290911456	102.519820674	37.562	-1.290472104	102.518054027	69.734
MG12	-1.860944171	102.278011136	22.789	-1.860497697	102.276282997	55.105
MG13	-1.483183819	102.677382622	37.555	-1.482729332	102.675620537	69.571
MG14	-1.138747570	102.445037819	43.964	-1.138317538	102.443265799	76.213
MG15	-1.188112735	102.562560401	22.858	-1.187674709	102.560786842	55.002
MG16	-1.245609725	102.859019342	4.406	-1.245153889	102.857238377	36.285
MG17	-1.038500832	102.011162856	9.406	-1.038097471	102.009400835	42.032
MG18	-1.814229442	102.683939828	50.714	-1.813763004	102.682194912	82.684
MG19	-1.401778162	102.656033321	34.663	-1.401327665	102.654267708	66.705
MG20	-1.946321386	102.021701939	11.167	-1.945885568	102.019987350	43.688

APPENDIX E: BURSA-WOLF-GENERATED DATASETS

This appendix contains datasets of virtual points for specific datums in actual parts of the world. The parameters of the best-fitting Bursa-Wolf transformations are known, and are given in ESRI (2012). However, the actual control points from which they were derived were not available for this study.

In each case, the totally-linear Bursa-Wolf transformation has been applied to 25-50 points that give good coverage of the geographical region in question. The purpose was to derive the Helmert model which gave the best fit at these virtual data points.

If the RMS fit of Helmert to the virtual points is small compared with the RMS fit of Bursa-Wolf to the original control points, then the pseudo-optimal Helmert transformation should give almost as good a fit as the optimal Bursa-Wolf. It can be regarded as the conformal transformation that comes closest to the optimal Bursa-Wolf. It will also provide an indication of how far the optimal Bursa-Wolf model is to being conformal.

E.1 Réunion Island

The datums for this dataset are Réunion 1947 and RGR 1992. From ESRI (2004), Réunion 1947 uses the International Ellipsoid and RGR 1992 uses GRS80.

28 virtual data points were selected as per Figure E-1. Apart from 4 points from the west and south coast, all are taken from 6′ intersections. Height values between 10m and 20m were assigned to the points. RGR 1992 coordinates (Cartesian, then geodetic) were estimated by applying the Bursa-Wolf transformation with the parameters given in ESRI (2012).

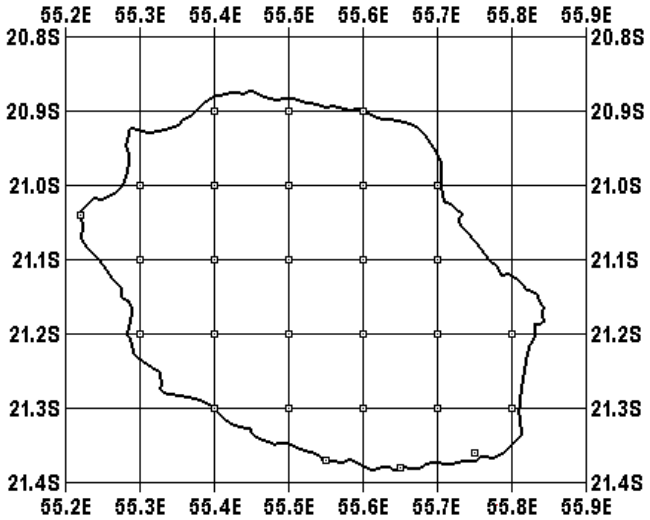


Figure E-1: Virtual data points in Réunion Island.

The geodetic coordinates are as follows.

	Réunion 1947 geodetic coordinates			RGR 1992 geodetic coordinates		
	ϕ (source) (degrees)	λ (source) (degrees)	h (source) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
MeshPt01	-20.90	55.40	15.00	-20.912467449	55.394091260	20.633
MeshPt02	-20.90	55.50	13.00	-20.912442122	55.494100320	16.985
MeshPt03	-20.90	55.60	17.00	-20.912416757	55.594109398	19.337
MeshPt04	-21.00	55.30	14.00	-21.012490878	55.294110831	21.375
MeshPt05	-21.00	55.40	16.00	-21.012465614	55.394119930	21.728
MeshPt06	-21.00	55.50	12.00	-21.012440314	55.494129029	16.081
MeshPt07	-21.00	55.60	18.00	-21.012414975	55.594138152	20.434

MeshPt08	-21.00	55.70	15.00	-21.012389599	55.694147271	15.787
MeshPt09	-21.10	55.30	13.00	-21.112489020	55.294139464	20.468
MeshPt10	-21.10	55.40	17.00	-21.112463782	55.394148609	22.822
MeshPt11	-21.10	55.50	14.00	-21.112438508	55.494157752	18.176
MeshPt12	-21.10	55.60	16.00	-21.112413196	55.594166912	18.530
MeshPt13	-21.10	55.70	12.00	-21.112387847	55.694176071	12.885
MeshPt14	-21.20	55.30	18.00	-21.212487165	55.294168117	25.560
MeshPt15	-21.20	55.40	15.00	-21.212461954	55.394177293	20.915
MeshPt16	-21.20	55.50	13.00	-21.212436705	55.494186480	17.271
MeshPt17	-21.20	55.60	17.00	-21.212411419	55.594195685	19.626
MeshPt18	-21.20	55.70	14.00	-21.212386096	55.694204888	14.982
MeshPt19	-21.20	55.80	16.00	-21.212360736	55.794214107	15.338
MeshPt20	-21.30	55.40	12.00	-21.312460128	55.394205986	18.009
MeshPt21	-21.30	55.50	18.00	-21.312434905	55.494215227	22.365
MeshPt22	-21.30	55.60	15.00	-21.312409646	55.594224464	17.722
MeshPt23	-21.30	55.70	13.00	-21.312384349	55.694233710	14.078
MeshPt24	-21.30	55.80	17.00	-21.312359015	55.794242975	16.435
ExtraPt25	-21.04	55.22	14.00	-21.052510310	55.214115000	22.729
ExtraPt26	-21.37	55.55	16.00	-21.382421031	55.544239982	19.610
ExtraPt27	-21.38	55.65	12.00	-21.392395596	55.644252132	13.977
ExtraPt28	-21.36	55.75	18.00	-21.372370647	55.744255658	18.315

E.2 Fatu Iva (Fatu Hiva)

The datums for this dataset are Fatu Iva 1972 and WGS 84. From ESRI (2004), Fatu Iva 1972 uses the International Ellipsoid.

25 virtual data points were selected as per Figure E-2. Apart from 7 points in coastal regions, all are taken from 0.02° intersections. Height values between 20m and 740m were assigned to the points. WGS84 coordinates (Cartesian, then geodetic) were estimated by applying the Bursa-Wolf transformation with the parameters given in ESRI (2012).

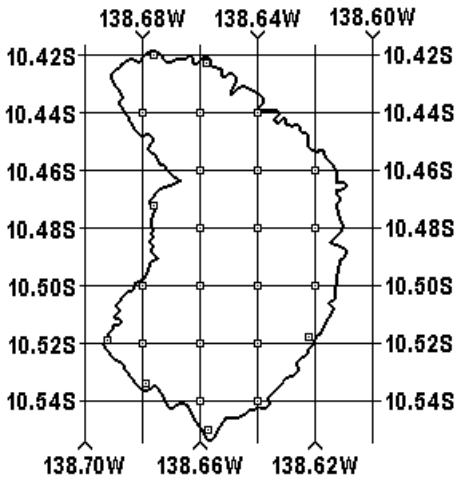


Figure E-2: Virtual data points in Fatu Iva.

The geodetic coordinates are as follows.

	Fatu Iva 1972 geodetic coordinates			WGS84 geodetic coordinates		
	ϕ (source) (degrees)	λ (source) (degrees)	h (source) (metres)	ϕ (target) (degrees)	λ (target) (degrees)	h (target) (metres)
MeshPt01	-10.440	-138.680	312.00	-10.396865666	-138.688988890	313.962
MeshPt02	-10.440	-138.660	356.00	-10.396868222	-138.668987092	357.771
MeshPt03	-10.440	-138.640	24.00	-10.396869464	-138.648985606	25.510
MeshPt04	-10.460	-138.660	115.00	-10.416871395	-138.668989745	115.870

MeshPt05	-10.460	-138.640	279.00	-10.416874374	-138.648987849	279.702
MeshPt06	-10.460	-138.620	98.00	-10.416876148	-138.628986239	98.469
MeshPt07	-10.480	-138.660	514.00	-10.436876810	-138.668991868	514.088
MeshPt08	-10.480	-138.640	636.00	-10.436879642	-138.648990010	635.912
MeshPt09	-10.480	-138.620	157.00	-10.436880375	-138.628988649	156.624
MeshPt10	-10.500	-138.680	55.00	-10.456876823	-138.688996458	54.346
MeshPt11	-10.500	-138.660	418.00	-10.456880496	-138.668994402	417.214
MeshPt12	-10.500	-138.640	737.00	-10.456884018	-138.648992383	736.074
MeshPt13	-10.500	-138.620	153.00	-10.456884384	-138.628991112	151.767
MeshPt14	-10.520	-138.680	165.00	-10.476881232	-138.688998820	163.510
MeshPt15	-10.520	-138.660	698.00	-10.476885500	-138.668996626	696.410
MeshPt16	-10.520	-138.640	683.00	-10.476887855	-138.648994886	681.208
MeshPt17	-10.540	-138.660	134.00	-10.496887556	-138.668999551	131.450
MeshPt18	-10.540	-138.640	28.00	-10.496889593	-138.648997889	25.231
ExtraPt19	-10.420	-138.676	35.00	-10.376861163	-138.684986317	37.728
ExtraPt20	-10.423	-138.658	28.00	-10.379863902	-138.666985104	30.418
ExtraPt21	-10.518	-138.622	49.00	-10.474887403	-138.630993588	46.997
ExtraPt22	-10.550	-138.657	30.00	-10.506889568	-138.666000604	26.973
ExtraPt23	-10.534	-138.679	33.00	-10.490883709	-138.688000560	30.877
ExtraPt24	-10.519	-138.692	26.00	-10.475879103	-138.700999863	24.647
ExtraPt25	-10.472	-138.676	35.00	-10.428871603	-138.684992690	35.501

APPENDIX F: CHEBYSHEV POLYNOMIALS

Chebyshev polynomials are a type of polynomial discovered by the Russian mathematician Pafnuty Chebyshev (1821-1894). Strictly speaking, there are two kinds of Chebyshev polynomial. However, all Chebyshev considered here are Chebyshev polynomials of the first kind and are denoted $T_n(x)$. (The "T" arises from several of the alternative transliterations of Chebyshev's name, which include Tchebyshev and Tchebycheff.)

Chebyshev polynomials are bounded by the limits -1 and 1 in the interval [-1, 1]. For n>0, $T_n(x) = \pm 1$ for n+1 values of x. Chebyshev polynomials are important in approximation theory because their roots are used as nodes in polynomial interpolation. Starting from the identities (F-1) and (F-2), they can be generated by the recurrence formula (F-3).

$$T_0(x) = 1; \quad \text{(F-1)}$$
$$T_1(x) = x; \quad \text{(F-2)}$$
$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x) \text{ for } n>1. \quad \text{(F-3)}$$

Among the low-degree Chebyshev polynomials are the following:

$$T_2(x) = 2x^2 - 1. \quad \text{(F-4)}$$
$$T_3(x) = 4x^3 - 3x. \quad \text{(F-5)}$$
$$T_4(x) = 8x^4 - 8x^2 + 1. \quad \text{(F-6)}$$
$$T_5(x) = 16x^5 - 20x^3 + 5x. \quad \text{(F-7)}$$

The graphs of the first 6 Chebyshev polynomials are given in Figures F1-1 to F1-4 below.

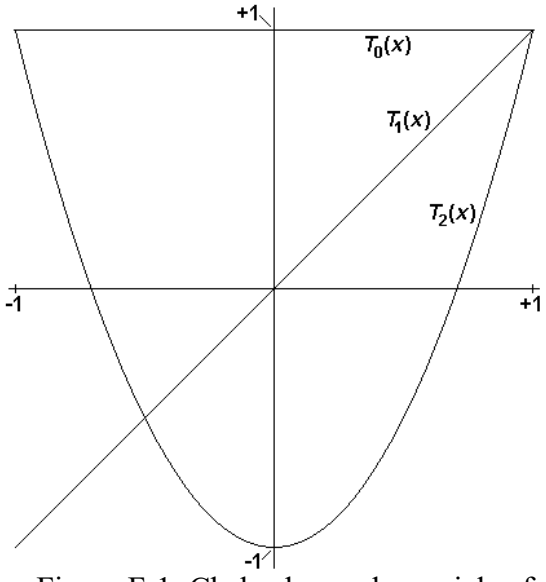


Figure F-1: Chebyshev polynomials of degree 0, 1 and 2.

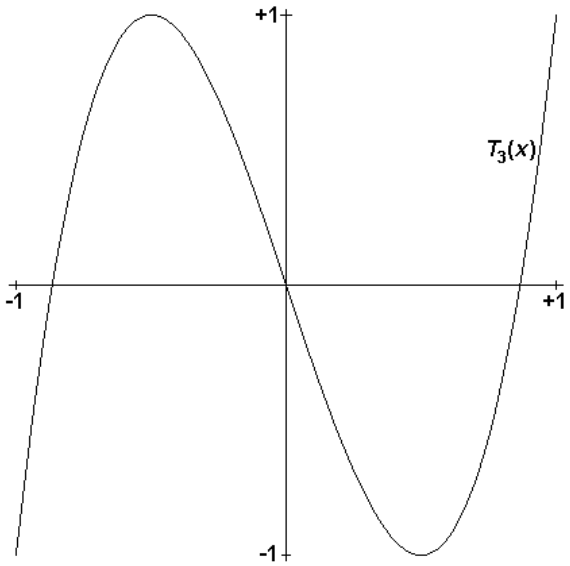


Figure F-2: Chebyshev polynomial of degree 3.

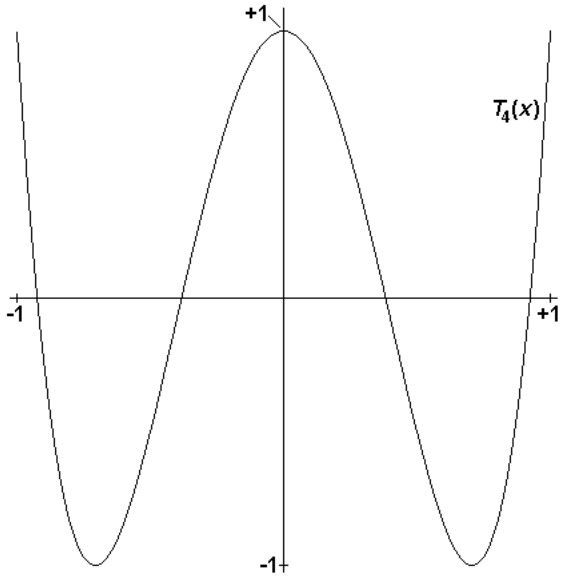


Figure F-3: Chebyshev polynomial of degree 4.

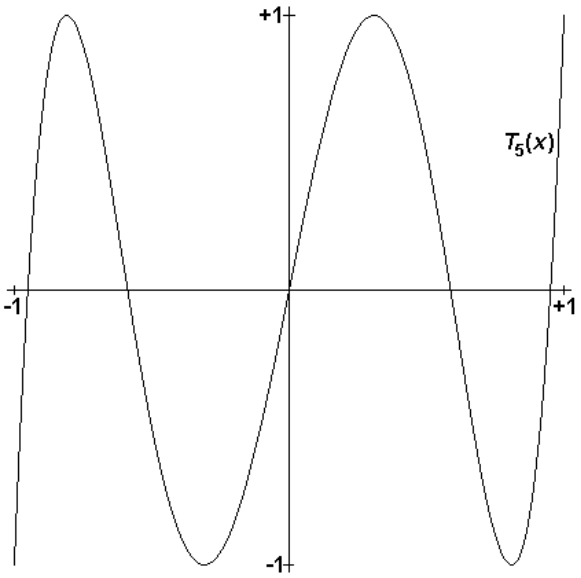


Figure F-4: Chebyshev polynomial of degree 5.

The resemblance of the curves to trigonometric functions is no mere coincidence. One of the properties of Chebyshev polynomials is that

$$T_n(x) = \cos(n \cdot \arccos x) \quad \text{if } |x| \leqslant 1. \tag{F-8}$$

APPENDIX G: KEY MACROS AND SUBROUTINES

This appendix lists a selection of key macros and subroutines written specifically for this research. The method of programming was predominantly Visual Basic for Applications (VBA) in Excel spreadsheets.

G.1 Key macros

This sub-appendix describes a selection of key macros written specifically for this research. These occurred in Excel spreadsheets and the emphasis here is on the interface with the user and how the user input is processed.

G.1.1 Application of models to geodetic coordinates

The scope of this macro consists of Standard Molodensky, Abridged Molodensky and the partially-conformal variations. It was used in the case study for Chapter 11, by which time the new variations had been fully developed. (The case studies for Chapters 8, 9 and 10 were conducted earlier with macros written separately as methods were evolving.) The interface with the user is as shown in Figure G-1.

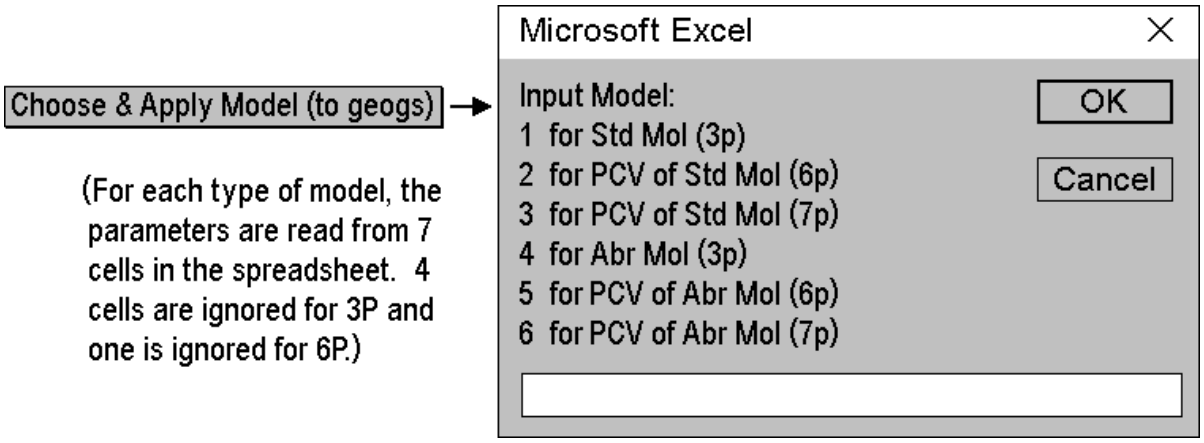


Figure G-1: Interface with the "Application of models to geodetic coordinates" macro.

When the transformation is applied to the data points, models 1 to 3 use the same 7-parameter formulae (2-70)-(2-72), but with ΔX_{ver}, ΔY_{ver}, ΔZ_{ver} set to ΔX_{hor}, ΔY_{hor}, ΔZ_{hor} if necessary and R_Z=0 if necessary. Models 4 to 6 use 7-parameter formulae (2-73)-(2-75) in the same way.

The outputs from the macro are:

- Latitude, longitude and ellipsoidal height computed from the model at all data points.
- Residuals in latitude, longitude and ellipsoidal height at all data points.
- RMSs in metres (latitude, longitude, ellipsoidal height, horizontal distance and 3D distance) for the set of data points.

G.1.2 Application of models to Cartesian coordinates

The scope of this macro consists of the transformations described in Sections 2.6, 2.8, 2.9, 2.11 and 2.12. It was used in the case study for Chapter 11, in the quest for a more unified approach than that used for the case studies for Chapters 8, 9 and 10 (where several macros were written over a long period). The interface with the user is as shown in Figure G-2.

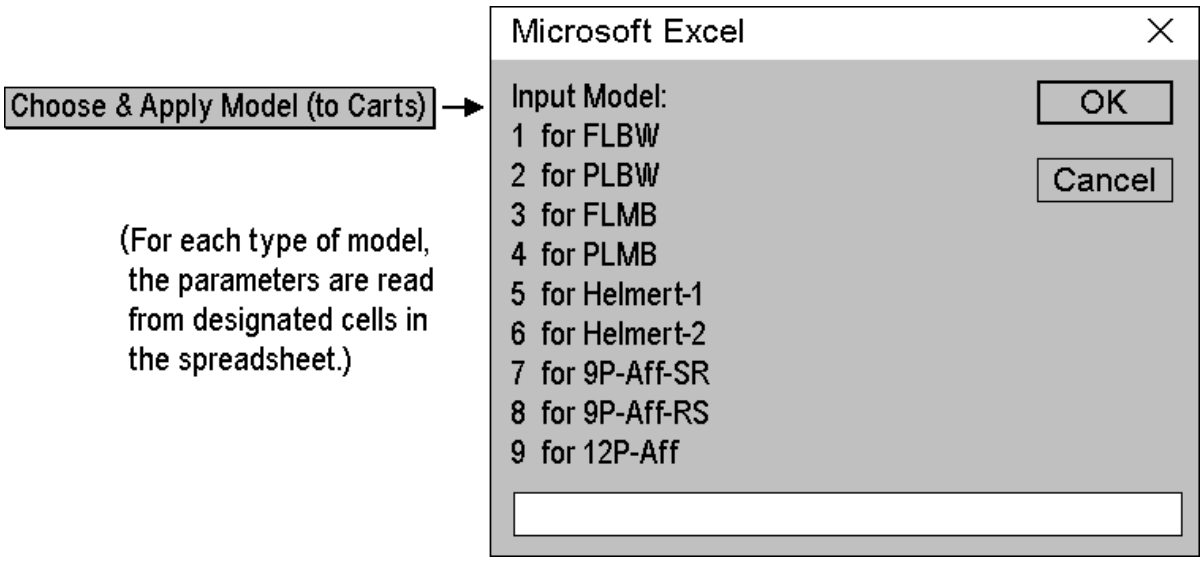


Figure G-2: Interface with the "Application of models to Cartesian coordinates" macro.

If, for example, the chosen model is 2 (for Partially-Linear Bursa-Wolf), the macro applies the following section of VBA code (via the subroutine ComputeShiftsFromModel) for each data point.

```
ElseIf iModel = 2 Then
    xA(i) = dx + s * (xs(i) - RZ * ys(i) + RY * zs(i))
    yA(i) = dy + s * (RZ * xs(i) + ys(i) - RX * zs(i))
    zA(i) = dz + s * (-RY * xs(i) + RX * ys(i) + zs(i))
```

The outputs from the macro are:

- Latitude, longitude and ellipsoidal height computed from the model at all data points.
- Residuals in latitude, longitude and ellipsoidal height at all data points.
- RMSs in metres (latitude, longitude, ellipsoidal height, horizontal distance and 3D distance) for the set of data points.

G.1.3 Application of MREs

This macro was written for the case studies in Chapters 8, 12 and 13. The interface with the user is as shown in Figure G-3.

If, for example, the chosen model is Ord5, the macro applies the following section of VBA code (via the subroutine ComputeShiftsFromModel) for each data point.

```
ElseIf Model = "Ord5" Then
    DLat = LatShift_OrdMRE_TP5(U1, V1)
    DLon = LonShift_OrdMRE_TP5(U1, V1)
```

The outputs from the macro are:

- Latitude and longitude computed from the model at all data points.
- Residuals in latitude and longitude at all data points.
- RMSs in metres (latitude, longitude, horizontal distance) for the set of control points.
- RMSs in metres (latitude, longitude, horizontal distance) for the set of test points if there is such a set.

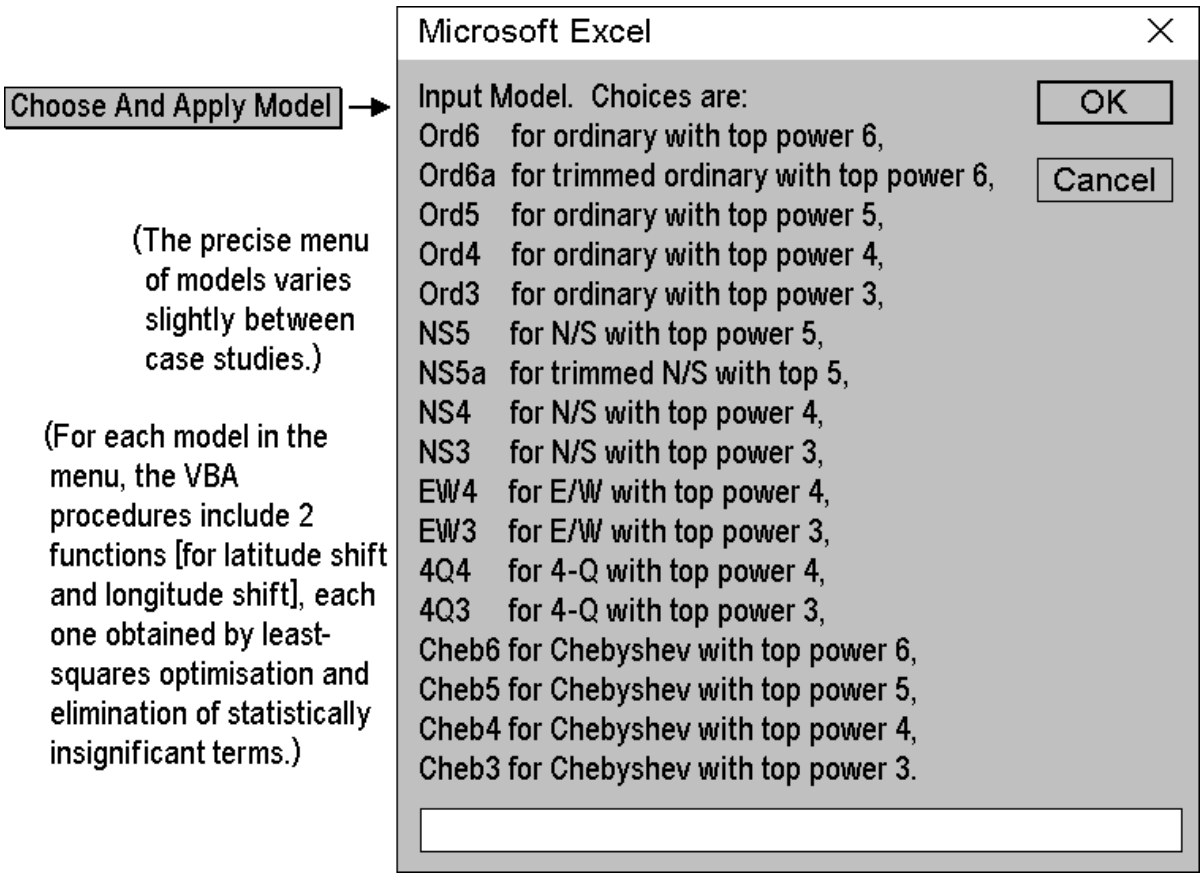


Figure G-3: Interface with the "Application of MREs" macro.

G.2 Key subroutines

This sub-appendix describes a selection of key subroutines written specifically for this research. The method of programming was predominantly Visual Basic for Applications (VBA) in Excel spreadsheets. VBA uses colour-coding for comments, but shading is used in the listings reproduced here.

The examples chosen are all related to the derivation of transformations. The first three are general and are used by a variety of macros. The 4th subroutine is a non-linear least-squares optimisation used in macros for deriving Helmert and affine transformations.

G.2.1 Forming and solving normal equations

The subroutine FormAndSolveNormalEqns was written to obtain the least-squares solution to a set of linear observation equations. It uses the well-known Cholesky decomposition to compute the inverse of the normal matrix, which is needed if standard errors are to be computed.

A positive definite matrix $\mathbf{M}$ can be decomposed into the product $\mathbf{LL^T}$ where $\mathbf{L}$ is lower-triangular. The method of computing the elements is forward substitution. The equations for $m_{1,1}$, $m_{2,1}, m_{2,2}$, $m_{3,1}$, $m_{3,2}$, $m_{3,3}$, … (taken in order) will produce solutions for the corresponding elements of $\mathbf{L}$. $\mathbf{L^{-1}}$ is obtained by solving $\mathbf{L} \cdot (\mathbf{col\ j\ of\ L^{-1}}) = (\mathbf{col\ j\ of\ I})$. $\mathbf{M^{-1}}$ can then be computed from $(\mathbf{L^{-1}})^{\mathbf{T}}\mathbf{L^{-1}}$.

The VBA code of the subroutine is as follows.

```
Sub FormAndSolveNormalEqns(m As Integer, n As Integer, _
                           A, B, W, X, V, SE, SEUW As Double)
'   VBA subroutine to form and solve normal equations for m linear
```

```
'   equations (AX = B + V) in n variables.  It is assumed that
'   m <= 4400 and n <= 100.
'   A is the design matrix which must have at least m rows and
'   n columns.  B is a column vector with at least m elements,
'   as is W (which consists of the weights).
'   OUTPUTS: X is the solution vector (at least n elements).
'            SE is the standard errors vector (at least n elements).
'            V is the residuals vector (at least m elements).
'            SEUW is the standard error of an observation of unit weight.
'   Author: Andrew Ruffhead.
    Dim ATW(1 To 100, 1 To 4400) As Double
    Dim ATWA(1 To 100, 1 To 100) As Double, ATWB(1 To 100) As Double
    Dim invATWA(1 To 100, 1 To 100) As Double, RHS(1 To 100) As Double
    Dim matL(1 To 100, 1 To 100) As Double
    Dim LatestCol(1 To 100)
    Dim UpSoln(1 To 100) As Double
    Dim i As Integer, j As Integer, k As Integer
    Dim iRow As Integer, jj As Integer, iRV as Integer
    Dim dummy As Double, Tol As Double
'   Compute ATW (= A transposed times the square version of W).
    For i = 1 To n
        For j = 1 To m
            ATW(i, j) = A(j, i) * W(j)
        Next j
    Next i
'   Compute the matrix ATWA and vector ATWB of the normal equations
'   (which take the form ATWA*X = ATWB).
    For i = 1 To n
        ATWB(i) = 0#
        For j = 1 To n
            ATWA(i, j) = 0#
            For k = 1 To m
                ATWA(i, j) = ATWA(i, j) + ATW(i, k) * A(k, j)
            Next k
        Next j
        For k = 1 To m
            ATWB(i) = ATWB(i) + ATW(i, k) * B(k)
        Next k
    Next i
'   Following the Cholesky method, the normal matrix ATWA is
'   decomposed into the product of a lower-triangular matrix
'   (matL) and its upper-triangular transpose.
'   Firstly, initialise lower-triangular matrix matL.
    For i = 1 To n
        For j = 1 To n
            matL(i, j) = 0#
        Next j
    Next i
'   Secondly, compute 1st row of lower-triangular matrix matL.
    If ATWA(1, 1) <= 0# Then
        MsgBox ("Argument of Square Root Not Positive." & vbLf & _
                "Subroutine abandoned.")
        Exit Sub
    End If
    matL(1, 1) = Sqr(ATWA(1, 1))
'   Thirdly, compute the rest of lower-triangular matrix matL,
'   one row at a time.
    For iRow = 2 To n
        For j = 1 To iRow - 1
            matL(iRow, j) = ATWA(iRow, j)
            For k = 1 To j - 1
                matL(iRow, j) = matL(iRow, j) - matL(iRow, k) * matL(j, k)
            Next k
            matL(iRow, j) = matL(iRow, j) / matL(j, j)
```

```
            Next j
            dummy = ATWA(iRow, iRow)
            For k = 1 To iRow - 1
                dummy = dummy - matL(iRow, k) ^ 2
            Next k
            If dummy <= 0# Then
                MsgBox ("Argument of Square Root Not Positive." & vbLf & _
                       "Subroutine abandoned.")
                Exit Sub
            End If
            matL(iRow, iRow) = Sqr(dummy)
        Next iRow
'       Solve matL*UpSoln=ATWB by forward-substitution.
        UpSoln(1) = ATWB(1) / matL(1, 1)
        For i = 2 To n
            UpSoln(i) = ATWB(i)
            For j = 1 To i - 1
                UpSoln(i) = UpSoln(i) - matL(i, j) * UpSoln(j)
            Next j
            UpSoln(i) = UpSoln(i) / matL(i, i)
        Next i
'       Solve matL(T)*X=UpSoln by back-substitution.
        X(n) = UpSoln(n) / matL(n, n)
        For i = n - 1 To 1 Step -1
            X(i) = UpSoln(i)
            For j = i + 1 To n
                X(i) = X(i) - matL(j, i) * X(j)
            Next j
            X(i) = X(i) / matL(i, i)
        Next i
'       Compute standard errors of the estimated parameters.
'       Compute invATWA, one column at a time.
        For jj = 1 To n
'           Set RHS to jjth column of identity matrix.
            For i = 1 To n
                RHS(i) = 0#
            Next i
            RHS(jj) = 1#
'           Solve matL*UpSoln=RHS by forward-substitution.
            UpSoln(1) = RHS(1) / matL(1, 1)
            For i = 2 To n
                UpSoln(i) = RHS(i)
                For j = 1 To i - 1
                    UpSoln(i) = UpSoln(i) - matL(i, j) * UpSoln(j)
                Next j
                UpSoln(i) = UpSoln(i) / matL(i, i)
            Next i
'           Solve matL(T)*LatestCol=UpSoln by back-substitution.
            LatestCol(n) = UpSoln(n) / matL(n, n)
            For i = n - 1 To 1 Step -1
                LatestCol(i) = UpSoln(i)
                For j = i + 1 To n
                    LatestCol(i) = LatestCol(i) - matL(j, i) * LatestCol(j)
                Next j
                LatestCol(i) = LatestCol(i) / matL(i, i)
            Next i
'           Copy LatestCol into jjth column of invATWA.
            For i = 1 To n
                invATWA(i, jj) = LatestCol(i)
            Next i
        Next jj
'       Compute provisional standard errors.
        For i = 1 To n
            SE(i) = Sqr(invATWA(i, i))
```

```
    Next i
'   Check product of ATWA and invATWA.
    Tol = 0.0001
    For i = 1 To n
        For j = 1 To n
            dummy = 0#
            For k = 1 To n
            dummy = dummy + ATWA(i, k) * invATWA(k, j)
            Next k
            If i = j Then
                If Abs(dummy - 1#) > Tol Then
                    iRV = MsgBox("Product of normal matrix and its " & _
                                 "inverse is suspect." & vbLf & _
                                 "Diagonal term is " & dummy & "." & _
                                 vbLf & "Continue with " & _
                                 "increased tolerance?", vbYesNo)
                    If iRV = vbYes Then
                        Tol = Abs(dummy - 1#)
                    Else
                        Exit Sub
                    End If
                End If
            Else
                If Abs(dummy) > Tol Then
                    iRV = MsgBox("Product of normal matrix and its " & _
                                 "inverse is suspect." & vbLf & _
                                 "Diagonal term is " & dummy & "." & _
                                 vbLf & "Continue with " & _
                                 "increased tolerance?", vbYesNo)
                    If iRV = vbYes Then
                        Tol = Abs(dummy)
                    Else
                        Exit Sub
                    End If
                End If
            End If
        Next j
    Next i
'   Compute residuals and standard error of an observation of unit weight.
'   Use the latter to correct the standard errors of the parameters.
    If n = m Then
        MsgBox "Note that there are no degrees of freedom," & vbLf & _
               "so there are no standard errors."
    Else
        SEUW = 0#
        For i = 1 To m
            V(i) = B(i)
            For j = 1 To n
                V(i) = V(i) - A(i, j) * X(j)
            Next j
            SEUW = SEUW + V(i) * W(i) * V(i)
        Next i
        SEUW = Sqr(SEUW / (m - n))
        For j = 1 To n
            SE(j) = SE(j) * SEUW
        Next j
    End If
End Sub
```

G.2.2 Solving a symmetric matrix equation by Cholesky

The subroutine SolveByCholesky was written to attempt a solution of symmetric matrix equations without computing an inverse matrix. The matrix is assumed to be positive definite, allowing a Cholesky decomposition $\mathbf{M=LL^T}$. A successful solution will result in the Boolean

variable Dunnit being set to True, and a solution check indicating how closely the solution fits the original equation. If the decomposition breaks down, the subroutine will

- display a message indicating where the breakdown occurred;
- exit the subroutine with the Boolean variable Dunnit set to False.

A positive definite matrix **M** can be decomposed into the product $\mathbf{LL^T}$ where **L** is lower-triangular. The method of computing the elements is forward substitution. The equations for $m_{1,1}$, $m_{2,1}, m_{2,2}$, $m_{3,1}$, $m_{3,2}$, $m_{3,3}$, … (taken in order) will produce solutions for the corresponding elements of **L**. The equation $\mathbf{Mx= LL^Tx=b}$. is split up into $\mathbf{L^Ty = b}$ (from which **y** is obtained by back-substitution) and **Lx=y**, from which **x** is obtained by forward substitution.

The VBA code of the subroutine is as follows.

```
Sub SolveByCholesky(SM() As Double, RHS() As Double, _
           n As Integer, Soln() As Double, Dunnit As Boolean)
'   Subroutine to solve matrix equation SM*Soln=RHS by Cholesky
'   decomposition.
'   SM is an array with a symmetric matrix in SM(1,1) to SM(n,n).
'   (Any other rows & columns in SM are ignored by this subroutine.)
'   RHS and Soln are n-vectors (or arrays with sufficiently-large
'   limits).
'   SM must be positive definite for Cholesky to work.  If Cholesky
'   fails, Dunnit will return as False.
'   ************************************************
'   *    Note that matL must be at least n*n in size   *
'   *    and RHSA must have at least n elements.       *
'   ************************************************
'   Author: Andrew Ruffhead.
    Dim matL(3123, 3123) As Double, RHSA(3123) As Double
    Dim i As Integer, j As Integer, k As Integer
    Dim misclos As Double, maxmisclos As Double
'   Initialise matL.
    For i = 1 To n
        For j = 1 To n
            matL(i, j) = 0#
        Next j
    Next i
'   Compute 1st row of matL.
    If SM(1, 1) <= 0# Then
        MsgBox ("Argument of 1st Square Root Not Positive." & vbLf & _
                "Subroutine abandoned.")
        MsgBox ("For row 1 in matrix L," & vbLf & _
                "the term to be square-rooted isn't positive." & _
                vbLf & "Subroutine SolveByCholesky abandoned.")
        Dunnit = False
        Exit Sub
    End If
    matL(1, 1) = Sqr(SM(1, 1))
'   Compute 1st row of matL.
    For i = 2 To n
'       Compute 1st row of matL.
        For j = 1 To i - 1
            matL(i, j) = SM(i, j)
            For k = 1 To j - 1
                matL(i, j) = matL(i, j) - matL(i, k) * matL(j, k)
            Next k
            matL(i, j) = matL(i, j) / matL(j, j)
        Next j
        dummy = SM(i, i)
```

```
        For k = 1 To i - 1
            dummy = dummy - matL(i, k) ^ 2
        Next k
        If dummy <= 0# Then
            MsgBox ("For row " & i & " in matrix L," & vbLf & _
                    "the term to be square-rooted isn't positive." & _
                    vbLf & "Subroutine SolveByCholesky abandoned.")
            Dunnit = False
            Exit Sub
        End If
        matL(i, i) = Sqr(dummy)
    Next i
'   Solve matL*RHSA=RHS by forward-substitution.
    RHSA(1) = RHS(1) / matL(1, 1)
    For i = 2 To n
        RHSA(i) = RHS(i)
        For j = 1 To i - 1
            RHSA(i) = RHSA(i) - matL(i, j) * RHSA(j)
        Next j
        RHSA(i) = RHSA(i) / matL(i, i)
    Next i
'   Solve matL(T)*Soln=RHSA by back-substitution.
    Soln(n) = RHSA(n) / matL(n, n)
    For i = n - 1 To 1 Step -1
        Soln(i) = RHSA(i)
        For j = i + 1 To n
            Soln(i) = Soln(i) - matL(j, i) * Soln(j)
        Next j
        Soln(i) = Soln(i) / matL(i, i)
    Next i
    MsgBox "Solution vector computed."
'   Check product of SM and Soln.
    maxmisclos = 0#
    For i = 1 To n
        misclos = -RHS(i)
        For j = 1 To n
            misclos = misclos + SM(i, j) * Soln(j)
        Next j
        misclos = Abs(misclos)
        If misclos > maxmisclos Then
            maxmisclos = misclos
        End If
    Next i
    MsgBox "Solution check: maximum misclosure = " _
           & maxmisclos & "."
    Dunnit = True
End Sub
```

G.2.3 Solving a matrix equation using MINVERSE

The subroutine SolveByTempInverse uses the Microsoft Excel MINVERSE function, which returns the inverse matrix for a given matrix. Both the input to and the output from MINVERSE are stored in an individual sheet of the Workbook using that function. If the matrix contains thousands of rows and columns, this creates considerable storage that may not be needed once the matrix equation is solved.

This subroutine

- writes the original matrix to a sheet called “TempCalcs”;
- creates an instruction to apply MINVERSE to the matrix and write the inverse below the original;
- solves the matrix equation by applying the inverse matrix;

- performs a solution check;
- clears the contents of "TempCalcs".

The VBA code of the subroutine is as follows.

```
Sub SolveByTempInverse(SM() As Double, RHS() As Double, _
    n As Integer, Soln() As Double, Dunnit As Boolean)
'   Subroutine to solve matrix equation SM*Soln=RHS by use of MINVERSE.
'   SM is an array with a symmetric matrix in SM(1,1) to SM(n,n).
'   (Any other rows & columns in SM are ignored by this subroutine.)
'   RHS and Soln are n-vectors (or arrays with sufficiently-large
'   limits).
'   If the subroutine reaches the finish, Dunnit will exit as True.
'   Author: Andrew Ruffhead.
    Dim i As Integer, j As Integer, k As Integer
    Dim misclos As Double, maxmisclos As Double
'   Write SM to TempCalcs.
    Worksheets("TempCalcs").Activate
    For i = 1 To n
        For j = 1 To n
            ActiveSheet.Cells(i, j) = SM(i, j)
        Next j
    Next i
'   Compute inverse below the original with 2 blank rows in between.
'   The formula is written into the block of cells of the destination
'   matrix (which will contain the inverse).  It is applied, so that the
'   computed inverse goes into Cells(n+3,1) to (2n+2,n).
    ActiveSheet.Range(Cells(n + 3, 1), _
                      Cells(2 * n + 2, n)).Select
    Selection.FormulaArray = "=MINVERSE(R[" & -n - 2 & _
                             "]C:R[-3]C[" & n - 1 & "])"
'   Compute the solution of the matrix equation.
    For i = 1 To n
        Soln(i) = 0#
        For j = 1 To n
            Soln(i) = Soln(i) + ActiveSheet.Cells(i + n + 2, j) * RHS(j)
        Next j
    Next i
    MsgBox "Solution vector computed."
'   Check product of SM and Soln.
    maxmisclos = 0#
    For i = 1 To n
        misclos = -RHS(i)
        For j = 1 To n
            misclos = misclos + SM(i, j) * Soln(j)
        Next j
        misclos = Abs(misclos)
        If misclos > maxmisclos Then
            maxmisclos = misclos
        End If
    Next i
    MsgBox "Solution check: maximum misclosure = " _
           & maxmisclos & "."
    Worksheets("TempCalcs").Cells.ClearContents
    Dunnit = True
End Sub
```

G.2.4 Least-squares optimisation of 6-parameter rigid transformation

The subroutine OptimiseRigidTrans was written to find the rigid transformation that gives a least-squares fit to Cartesian coordinates in two datums. Note that it uses the subroutine in sub-appendix G.2.1.

The VBA code of the subroutine is as follows.

```
Sub OptimiseRigidTrans(nStns As Integer, xs, ys, zs, xt, yt, zt, _
                       Param, Res)
'   VBA Subroutine to find the rigid transformation that gives a
'   least-squares fit to Cartesian coordinates in two datums.
'   It is assumed that the number of stations (nStns) does not
'   exceed 100, hence the internal rows-limit of 300.
'   The source arrays (xs, ys & zs) and the target arrays (xt, yt &
'   zt) must be given at least nStns elements by the calling module.
'   Note: it is important that any scale change is removed from the
'   Cartesian coordinates either by the source coordinates being
'   scaled or the target coordinates being "unscaled".
'   OUTPUTS: Param is the parameter vector of at least 6 elements
'                   (the first 3 are shifts in metres; the
'                    next 3 are position-vector rotations in radians).
'            Res is the residuals vector (at least 3*nStns
'                   elements: 3i-2, 3i-1 & 3i apply to stn i).
'   Author: Andrew Ruffhead.
    Dim i As Integer
    Dim j As Integer, nObs As Integer
    Dim MatA(300, 6) As Double, VecB(300) As Double
    Dim VecW(300) As Double
    Dim SE(6) As Double, SEUW As Double
    Dim sx As Double, sy As Double, sz As Double
    Dim cx As Double, cy As Double, cz As Double
    Dim rx1 As Double, ry1 As Double, rz1 As Double
    Dim LargCor As Double
'   Set number of observations.
    nObs = 3 * nStns
'   Form the matrix (MatA) and vectors (VecB, VecW) which
'   feature in the observation equations.
    For i = 1 To nStns
        MatA(3 * i - 2, 1) = 1#
        MatA(3 * i - 2, 2) = 0#
        MatA(3 * i - 2, 3) = 0#
        MatA(3 * i - 2, 4) = 0#
        MatA(3 * i - 2, 5) = zs(i)
        MatA(3 * i - 2, 6) = -ys(i)
        VecB(3 * i - 2) = xt(i) - xs(i)
        MatA(3 * i - 1, 1) = 0#
        MatA(3 * i - 1, 2) = 1#
        MatA(3 * i - 1, 3) = 0#
        MatA(3 * i - 1, 4) = -zs(i)
        MatA(3 * i - 1, 5) = 0#
        MatA(3 * i - 1, 6) = xs(i)
        VecB(3 * i - 1) = yt(i) - ys(i)
        MatA(3 * i, 1) = 0#
        MatA(3 * i, 2) = 0#
        MatA(3 * i, 3) = 1#
        MatA(3 * i, 4) = ys(i)
        MatA(3 * i, 5) = -xs(i)
        MatA(3 * i, 6) = 0#
        VecB(3 * i) = zt(i) - zs(i)
    Next i
    For j = 1 To nObs
        VecW(j) = 1#
    Next j
'   Do the first least-squares optimisation.
    Call FormAndSolveNormalEqns(nObs, 6, MatA, VecB, VecW, _
                                Param, Res, SE, SEUW)
'   Store the rotation parameters in rx1, ry1 & rz1.  The next
```

```
'   least-squares optimisation will compute corrections to shifts
'   0, 0, 0 and rotations rx1, ry1 & rz1.
    rx1 = Param(4)
    ry1 = Param(5)
    rz1 = Param(6)
'   Calculate the trig ratios of the estimated rotations.
    sx = Sin(rx1)
    sy = Sin(ry1)
    sz = Sin(rz1)
    cx = Cos(rx1)
    cy = Cos(ry1)
    cz = Cos(rz1)
'   Form the matrix (MatA) and vectors (VecB, VecW) which
'   feature in the observation equations.
    For i = 1 To nStns
        MatA(3 * i - 2, 1) = 1#
        MatA(3 * i - 2, 2) = 0#
        MatA(3 * i - 2, 3) = 0#
        MatA(3 * i - 2, 4) = (cx * sy * cz + sx * sz) * ys(i) _
                           + (cx * sz - sx * sy * cz) * zs(i)
        MatA(3 * i - 2, 5) = -sy * cz * xs(i) + sx * cy * cz * ys(i) _
                                              + cx * cy * cz * zs(i)
        MatA(3 * i - 2, 6) = -cy * sz * xs(i) _
                           - (sx * sy * sz + cx * cz) * ys(i) _
                           + (sx * cz - cx * sy * sz) * zs(i)
        VecB(3 * i - 2) = xt(i) - (cy * cz * xs(i) _
                        + (sx * sy * cz - cx * sz) * ys(i) _
                        + (sx * sz + cx * sy * cz) * zs(i))
        MatA(3 * i - 1, 1) = 0#
        MatA(3 * i - 1, 2) = 1#
        MatA(3 * i - 1, 3) = 0#
        MatA(3 * i - 1, 4) = (cx * sy * sz - sx * cz) * ys(i) _
                           - (cx * cz + sx * sy * sz) * zs(i)
        MatA(3 * i - 1, 5) = -sy * sz * xs(i) _
                           + sx * cy * sz * ys(i) + cx * cy * sz * zs(i)
        MatA(3 * i - 1, 6) = cy * cz * xs(i) _
                           + (sx * sy * cz - cx * sz) * ys(i) _
                           + (cx * sy * cz + sx * sz) * zs(i)
        VecB(3 * i - 1) = yt(i) - (cy * sz * xs(i) _
                        + (cx * cz + sx * sy * sz) * ys(i) _
                        + (cx * sy * sz - sx * cz) * zs(i))
        MatA(3 * i, 1) = 0#
        MatA(3 * i, 2) = 0#
        MatA(3 * i, 3) = 1#
        MatA(3 * i, 4) = cx * cy * ys(i) - sx * cy * zs(i)
        MatA(3 * i, 5) = -cy * xs(i) - sx * sy * ys(i) - cx * sy * zs(i)
        MatA(3 * i, 6) = 0#
        VecB(3 * i) = zt(i) - (-sy * xs(i) + sx * cy * ys(i) _
                    + cx * cy * zs(i))
    Next i
'   Do the second least-squares optimisation.
    Call FormAndSolveNormalEqns(nObs, 6, MatA, VecB, VecW, _
                                Param, Res, SE, SEUW)
'   Compute the largest correction to rotations.
    LargCor = Param(4)
    If Abs(Param(5)) > Abs(LargCor) Then
        LargCor = Param(5)
    End If
    If Abs(Param(6)) > Abs(LargCor) Then
        LargCor = Param(6)
    End If
    LargCor = LargCor * 206264.8
'   Apply corrections to the approximate rotations.
    Param(4) = rx1 + Param(4)
```

```
    Param(5) = ry1 + Param(5)
    Param(6) = rz1 + Param(6)
'   Display closing message.
    MsgBox "Rigid transformation computed OK." & vbLf & vbLf & _
           "Largest correction to rotations" & vbLf & _
           "was " & LargCor & " arc-seconds."
End Sub
```

APPENDIX H: MULTIPLE REGRESSION EQUATIONS FROM CASE STUDIES

This appendix contains multiple regression equations (MREs) derived during the case studies recorded in Chapters 8, 12 and 13. The MREs were obtained by eliminating ratios lower than one (ERLTO) except where the word "trimmed" indicates that further elimination of terms occurred.

H.1 Western Australia (AGD84 to GDA94)

The normalised intermediate coordinates U and V were defined as follows:

$$U = 0.09298\left(\phi_{\text{in deg}} + 24.350\right); \tag{H-1}$$

$$V = 0.12425\left(\lambda_{\text{in deg}} - 120.949\right). \tag{H-2}$$

The Ordinary MREs with top power 3 (AGD84→GDA94) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & 4.84733 + 0.27367U + 0.30298V - 0.07386U^2 - \\ & 0.17775UV - 0.01670V^2 - 0.13733U^2V - \\ & 0.01533V^3 + 0.06035U^3V + 0.10983U^2V^2 + \\ & 0.08154UV^3 + 0.09873U^3V^2 + 0.15702U^3V^3\end{aligned} \tag{H-3}$$

$$\begin{aligned}\Delta\lambda('') = {} & 4.90745 - 0.47032U - 0.08276V + 0.11823U^2 - \\ & 0.06109UV - 0.08940V^2 + 0.10322U^3 - 0.04413U^2V - \\ & 0.07009UV^2 + 0.12713U^3V + 0.05920U^2V^2 + \\ & 0.23763UV^3 - 0.33383U^3V^3\end{aligned} \tag{H-4}$$

The Ordinary MREs with top power 4 (AGD84→GDA94) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & 4.85578 + 0.27528U + 0.31758V - 0.10860U^2 - \\ & 0.14970UV - 0.08923V^2 - 0.17683U^2V - 0.07290UV^2 - \\ & 0.04784V^3 + 0.04003U^4 + 0.29636U^2V^2 + 0.08413V^4 + \\ & 0.26513U^3V^2 + 0.05003U^2V^3 + 0.12900UV^4 - \\ & 0.12619U^4V^2 - 0.11908U^2V^4 - 0.26421U^3V^4\end{aligned} \tag{H-5}$$

$$\begin{aligned}\Delta\lambda('') = {} & 4.90452 - 0.50396U - 0.07512V + 0.14004U^2 + \\ & 0.06001UV - 0.09021V^2 + 0.22806U^3 + 0.05712U^4 - \\ & 0.45726U^3V + 0.06987UV^3 - 0.52469U^4V + \\ & 0.42484U^3V^2 - 0.53170U^2V^3 + 0.84934U^4V^2 - \\ & 0.28994U^3V^3 + 0.47094U^2V^4 + 0.52270U^4V^2 - \\ & 0.82683U^4V^4\end{aligned} \tag{H-6}$$

The Ordinary MREs with top power 5 (AGD84→GDA94) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & 4.85558 + 0.28867U + 0.35309V - 0.09081U^2 - \\ & 0.24520UV - 0.08930V^2 - 0.03469U^3 - 0.47740U^2V - \\ & 0.49411UV^2 - 0.17675V^3 + 0.77465UV^3 + 0.08823V^4 + \\ & 2.70969U^3V^2 + 1.36942U^2V^3 + 0.64534UV^4 + \\ & 0.09572V^5 - 0.32400U^5V + 3.03422U^4V^2 - \\ & 2.91907U^3V^3 - 0.77424UV^5 + 0.51829U^5V^2 - \\ & 5.72298U^4V^3 - 2.22806U^3V^4 - 0.91912U^2V^5 - \\ & 1.96556U^5V^3 + 2.27936U^3V^5 + 2.20019U^5V^4 + \\ & 2.75084U^4V^5\end{aligned} \tag{H-7}$$

$$\begin{aligned}\Delta\lambda('') = {} & 4.90685 - 0.52358U - 0.08176V + 0.19708UV - \\ & 0.08603V^2 + 0.21648U^3 + 0.51775U^2V + 0.07427V^3 + \\ & 0.63193U^4 - 1.63369U^3V + 0.59289U^2V^2 - \\ & 0.40728UV^3 + 0.51539U^5 - 3.48978U^4V + \\ & 2.74251U^3V^2 - 3.50038U^2V^3 - 0.08505V^5 - \\ & 1.21612U^5V + 3.05007U^4V^2 + 2.78273U^3V^3 - \\ & 0.46298U^2V^4 + 0.33046UV^5 + 8.15979U^4V^3 - \\ & 4.0655U^3V^4 + 3.07087U^2V^5 - 8.46034U^4V^4 - \\ & 4.5479U^5V^4 + 5.20562U^5V^5\end{aligned} \quad \text{(H-8)}$$

The North/South MREs with top power 3 (AGD84→GDA94) are as follows:

$$\Delta\phi('') = \begin{cases} 4.84275 + 0.21621U + 0.31687V - 0.26613UV - \\ 0.00933V^2 - 0.03400V^3 + 0.29052UV^3 \\ + 0.50367U^2 - 0.76097U^2V - 0.88740U^3 + \\ 1.94070U^3V - 0.95665U^3V^2 & \text{if } U>0 \\ -0.23500U^2 - 0.70577U^2V + 1.14841U^2V^3 - \\ 0.12447U^3 - 0.4854U^3V + 0.92071U^3V^3 & \text{if } U\le 0 \end{cases} \quad \text{(H-9)}$$

$$\Delta\lambda('') = \begin{cases} 4.90292 - 0.54841U - 0.07652V + 0.15695UV - \\ 0.08494V^2 \\ + 0.37819U^2 - 0.66264U^2V + 0.35811U^2V^3 + \\ 0.90385U^3V^2 - 0.90846U^3V^3 & \text{if } U>0 \\ -0.06008U^2 + 0.63082U^2V + 0.16441U^2V^2 - \\ 0.92301U^2V^3 + 0.63091U^3V - 1.13657U^3V^3 & \text{if } U\le 0 \end{cases} \quad \text{(H-10)}$$

The East/West MREs with top power 3 (AGD84→GDA94) are as follows:

$$\Delta\phi('') = \begin{cases} 4.85440 + 0.28108U - 0.06837U^2 + 0.31860V - \\ 0.36983UV - 0.18860U^2V + 0.17401U^3V \\ -0.12831V^2 + 0.43565UV^2 + 0.15272U^2V^2 + \\ 0.07748V^3 - 0.16418UV^3 - 0.19082U^3V^3 & \text{if } V>0 \\ -0.12333V^2 - 0.94440UV^2 + 0.77207U^3V^2 - \\ 0.16903V^3 - 0.82611UV^3 + 0.59531U^3V^3 & \text{if } V\le 0 \end{cases} \quad \text{(H-11)}$$

$$\Delta\lambda('') = \begin{cases} 4.90613 - 0.48497U + 0.12410U^2 + 0.13736U^3 - \\ 0.08120V - 0.09352UV + 0.30683U^3V \\ -0.08896V^2 + 0.19489UV^2 - 0.94433U^3V^2 + \\ 0.45110U^3V^3 & \text{if } V>0 \\ -0.08847V^2 + 0.33033UV^3 + 0.34610U^2V^3 - \\ 0.77107U^3V^3 & \text{if } V\le 0 \end{cases} \quad \text{(H-12)}$$

The Four-Quadrant MREs with top power 3 (AGD84→GDA94) are as follows:

$$\Delta\phi('') = \begin{cases} 4.85208 + 0.27734U + 0.28524V - \\ 0.06942U^2 - 0.30004UV - 0.02097V^3 + \\ 0.22313UV^2 - 0.07558U^2V + 0.15936U^3V \\ -0.13175U^3V^2 & \rightarrow \text{ if } U>0 \ \&\ V>0 \\ -45.19674U^2V^2 - 36.30299U^2V^3 + \\ 109.81505U^3V^2 + 60.82490U^3V^3 & \text{if } U>0 \ \&\ V\le 0 \\ -0.27203U^2V^2 + 0.46726U^2V^3 & \rightarrow \text{ if } U\le 0 \ \&\ V>0 \\ +1.32051U^2V^2 + 0.21603U^2V^3 + 0.94090U^3V^2 & \rightarrow \text{ if } U\le 0 \ \&\ V\le 0 \end{cases} \quad \text{(H-13)}$$

$$\begin{aligned}\Delta\lambda('') = {} & 4.91183 - 0.49374U - 0.07471V + 0.07927U^2 - \\ & 0.04631UV - 0.10209V^2 + 0.11307U^3 + \\ & 0.2287U^2V - 0.17939UV^2 + 0.28894UV^3 + \\ & 0.48513U^3V^3 \\ & -1.37939U^3V^2 + 0.6043U^3V^3 \rightarrow \text{ if } U>0 \ \&\ V>0\end{aligned} \quad \text{(H-14)}$$

$$
\left.\begin{array}{l} -16.99731U^2V^2 - 23.68164U^2V^3 + \\ 58.29948U^3V^2 + 68.89665U^3V^3 \end{array}\right\} \quad \text{if } U>0 \ \& \ V\leq 0
$$

$$
\left.\begin{array}{l} -3.88194U^2V^2 + 3.54316U^2V^3 - 5.54656U^3V^2 + \\ 4.7573U^3V^3 \end{array}\right\} \quad \text{if } U\leq 0 \ \& \ V>0
$$

$$
-0.55215U^3V^3 \quad \rightarrow \quad \text{if } U\leq 0 \ \& \ V\leq 0
$$

The Chebyshev MREs with top power 3 (AGD84→GDA94) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 4.81666 + 0.28945T_1(U) + 0.24338T_1(V) - 0.02357T_2(U) - \\
& 0.11737T_1(U)T_1(V) + 0.00663T_2(U) + 0.00679T_3(U) - \\
& 0.04619T_2(U)T_1(V) + 0.01686T_1(U)T_2(V) + \\
& 0.01413T_2(U)T_2(V) + 0.00716T_3(U)T_2(V) + \\
& 0.00427T_2(U)T_3(V) - 0.00577T_3(U)T_3(V)
\end{aligned} \tag{H-15}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 4.94727 - 0.41158T_1(U) - 0.11958T_1(V) + 0.08546T_2(U) - \\
& 0.02455T_2(V) + 0.02934T_3(U) - 0.03755T_2(U)T_1(V) - \\
& 0.02833T_1(U)T_2(V) - 0.03631T_3(U)T_1(V) + \\
& 0.01940T_2(U)T_2(V) - 0.01921T_3(U)T_3(V)
\end{aligned} \tag{H-16}
$$

The Chebyshev MREs with top power 4 (AGD84→GDA94) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 4.83757 + 0.32101T_1(U) + 0.21223T_1(V) - \\
& 0.16269T_1(U)T_1(V) + 0.01576T_2(V) + 0.01570T_3(U) - \\
& 0.07898T_2(U)T_1(V) + 0.02747T_1(U)T_2(V) - 0.00479T_3(V) + \\
& 0.00531T_4(U) - 0.01419T_3(U)T_1(V) + 0.01864T_2(U)T_2(V) + \\
& 0.00523T_4(V) - 0.00841T_4(U)T_1(V) + 0.00724T_3(U)T_2(V) - \\
& 0.00599T_3(U)T_3(V) - 0.00767T_4(U)T_3(V) + 0.00508T_4(U)T_4(V)
\end{aligned} \tag{H-17}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.08523 - 0.16504T_1(U) - 0.32962T_1(V) + 0.24866T_2(U) - \\
& 0.39929T_1(U)T_1(V) + 0.07189T_2(V) + 0.11351T_3(U) - \\
& 0.27427T_2(U)T_1(V) + 0.15707T_1(U)T_2(V) - 0.01603T_3(V) + \\
& 0.02313T_4(U) - 0.17005T_3(U)T_1(V) + 0.11695T_2(U)T_2(V) - \\
& 0.03044T_1(U)T_3(V) - 0.01053T_4(V) - 0.02142T_4(U)T_1(V) + \\
& 0.05239T_3(U)T_2(V) - 0.01571T_3(U)T_3(V) - \\
& 0.02308T_2(U)T_4(V) + 0.01323T_4(U)T_3(V) - \\
& 0.01185T_4(U)T_4(V)
\end{aligned} \tag{H-18}
$$

The Chebyshev MREs with top power 5 (AGD84→GDA94) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 5.32804 + 1.24825T_1(U) - 0.64396T_1(V) + 0.65181T_2(U) - \\
& 1.74046T_1(U)T_1(V) + 0.56118T_2(V) + 0.41132T_3(U) - \\
& 1.21532T_2(U)T_1(V) + 1.07474T_1(U)T_2(V) - 0.20089T_3(V) + \\
& 0.16740T_4(U) - 0.68166T_3(U)T_1(V) + 0.74079T_2(U)T_2(V) - \\
& 0.37953T_1(U)T_3(V) + 0.01307T_4(V) + 0.05518T_5(U) - \\
& 0.29188T_4(U)T_1(V) + 0.45055T_3(U)T_2(V) - 0.25078T_2(U)T_3(V) + \\
& 0.02988T_1(U)T_4(V) + 0.04782T_5(V) - 0.09314T_5(U)T_1(V) + \\
& 0.18013T_4(U)T_2(V) - 0.15238T_3(U)T_3(V) + 0.06356T_1(U)T_5(V) + \\
& 0.07097T_5(U)T_2(V) - 0.06480T_4(U)T_3(V) + 0.06653T_2(U)T_5(V) - \\
& 0.0263T_5(U)T_3(V) + 0.03931T_3(U)T_5(V) + 0.01037T_5(U)T_4(V) + \\
& 0.02342T_4(U)T_5(V)
\end{aligned} \tag{H-19}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 4.55106 - 1.20881T_1(U) + 0.80082T_1(V) - 0.44992T_2(U) + \\
& 1.71129T_1(U)T_1(V) - 1.03511T_2(V) - 0.36942T_3(U) + \\
& 1.22676T_2(U)T_1(V) - 1.93790T_1(U)T_2(V) + 0.82344T_3(V) - \\
& 0.13523T_4(U) + 0.78620T_3(U)T_1(V) - 1.35290T_2(U)T_2(V) + \\
& 1.59425T_1(U)T_3(V) - 0.44548T_4(V) - 0.08177T_5(U) + \\
& 0.34312T_4(U)T_1(V) - 0.91217T_3(U)T_2(V) + \\
& 1.11000T_2(U)T_3(V) - 0.77843T_1(U)T_4(V) + 0.09346T_5(V) + \\
& 0.14460T_5(U)T_1(V) - 0.35611T_4(U)T_2(V) + \\
& 0.72889T_3(U)T_3(V) - 0.59404T_2(U)T_4(V) +
\end{aligned} \tag{H-20}
$$

$$0.22950T_1(U)T_5(V) - 0.15325T_5(U)T_2(V) + 0.27529T_4(U)T_3(V) - 0.33570T_3(U)T_4(V) + 0.10236T_2(U)T_5(V) + 0.11485T_5(U)T_3(V) - 0.14424T_4(U)T_4(V) + 0.10749T_3(U)T_5(V) - 0.04274T_5(U)T_4(V) + 0.02077T_5(U)T_5(V)$$

H.2 Slovenia (D48 to D48)

The normalised intermediate coordinates U and V were defined as follows:

$$U = 1.37174\left(\phi_{\text{in deg}} - 46.150\right); \quad \text{(H-21)}$$

$$V = 0.62035\left(\lambda_{\text{in deg}} - 14.984\right). \quad \text{(H-22)}$$

The Ordinary MREs with top power 3 (D48→D96) as follows:

$$\begin{aligned}\Delta\phi('') = {} & -1.09110 - 0.26191U + 0.14379V + 0.01043U^2 - \\ & 0.04808UV - 0.01490U^3 - 0.02248UV^2 + 0.01086V^3 - \\ & 0.04603U^2V^2 - 0.00727UV^3 + 0.02513U^3V^2 - \\ & 0.01861U^2V^3 + 0.06285U^3V^3\end{aligned} \quad \text{(H-23)}$$

$$\begin{aligned}\Delta\lambda('') = {} & -17.24896 - 0.12073U - 0.67593V + 0.02208U^2 + \\ & 0.03339UV - 0.02324V^2 + 0.00452U^3 - 0.06066UV^2 - \\ & 0.03205U^3V - 0.01233U^2V^2 - 0.09246UV^3 - 0.05351U^3V^2 + \\ & 0.08356U^2V^3 + 0.10639U^3V^3\end{aligned} \quad \text{(H-24)}$$

The Ordinary MREs with top power 4 (D48→D96) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & -1.09254 - 0.25883U + 0.13974V + 0.02893U^2 - \\ & 0.02371UV + 0.01423V^2 - 0.02433U^3 + \\ & 0.04846U^2V - 0.04976UV^2 + 0.02588V^3 - \\ & 0.03018U^4 - 0.05885U^3V - 0.24328U^2V^2 - \\ & 0.10192UV^3 - 0.01228V^4 - 0.10423U^4V + \\ & 0.09507U^3V^2 - 0.05237U^2V^3 - 0.01024UV^4 + \\ & 0.34098U^4V^2 + 0.16784U^3V^3 + 0.40514U^2V^4 + \\ & 0.21526U^4V^3 - 0.15746U^3V^4 - 0.56827U^4V^4\end{aligned} \quad \text{(H-25)}$$

$$\begin{aligned}\Delta\lambda('') = {} & -17.24763 - 0.12702U - 0.66899V - 0.00388U^2 - \\ & 0.00886UV - 0.02962V^2 + 0.01851U^3 - 0.02959U^2V - \\ & 0.05122V^3 + 0.04077U^4 + 0.11865U^3V + 0.17691U^2V^2 + \\ & 0.13323UV^3 - 0.03941V^4 + 0.10803U^4V - \\ & 0.28827U^3V^2 + 0.13186U^2V^3 + 0.08709UV^4 - \\ & 0.36950U^4V^2 - 0.53553U^3V^3 - 0.31043U^2V^4 + \\ & 0.93646U^4V^4\end{aligned} \quad \text{(H-26)}$$

The Ordinary MREs with top power 5 (D48→D96) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & -1.09239 - 0.25651U + 0.13722V + \\ & 0.00398U^2 - 0.00928UV + 0.03583V^2 - \\ & 0.06154U^3 + 0.11511U^2V - 0.11715UV^2 + \\ & 0.06224V^3 + 0.03207U^4 - 0.09360U^3V - \\ & 0.16669U^2V^2 - 0.18192UV^3 - 0.10907V^4 + \\ & 0.07648U^5 - 0.28498U^4V + 0.48608U^3V^2 - \\ & 0.50455U^2V^3 + 0.32613UV^4 - 0.12733V^5 - \\ & 0.05196U^5V + 0.19303U^4V^2 + 0.27643U^3V^3 + \\ & 0.27463U^2V^4 + 0.37929UV^5 - 0.32722U^5V^2 + \\ & 1.15495U^4V^3 - 1.39624U^3V^4 + 0.46972U^2V^5 - \\ & 0.18176U^5V^3 + 0.37967U^4V^4 - 0.87756U^3V^5 + \\ & 0.13584U^4V^5\end{aligned} \quad \text{(H-27)}$$

$$\begin{aligned}\Delta\lambda('') = &-17.24846 - 0.12596U - 0.67449V - \\ &0.04157UV - 0.01873V^2 + 0.03764U^3 + \\ &0.06088U^2V - 0.11253UV^2 - 0.02280V^3 + \\ &0.01962U^4 + 0.30531U^3V + 0.48338U^2V^2 + \\ &0.28615UV^3 - 0.10173V^4 - 0.04572U^5 - \\ &0.03053U^4V - 0.29896U^3V^2 - 0.14922U^2V^3 + \\ &0.60439UV^4 - 0.08913V^5 - 0.27757U^5V - \\ &0.66745U^4V^2 - 1.70801U^3V^3 - 1.66776U^2V^4 + \\ &0.26339UV^5 + 0.32578U^5V^2 + 0.46713U^3V^4 - \\ &0.75765U^2V^5 + 1.84396U^5V^3 + 1.88527U^4V^4 + \\ &2.23054U^3V^5 - 0.53145U^5V^4 + 0.51841U^4V^5 - \\ &2.91632U^5V^5\end{aligned} \qquad \text{(H-28)}$$

The Ordinary MREs with top power 6 (D48→D96) are as follows:

$$\begin{aligned}\Delta\phi('') = &-1.09199 - 0.25671U + 0.13307V - 0.01039U^2 + \\ &0.02238UV + 0.01904V^2 - 0.03367U^3 + \\ &0.17596U^2V - 0.21032UV^2 + 0.10141V^3 + \\ &0.11755U^4 - 0.42027U^3V + 0.28201U^2V^2 - \\ &0.45957UV^3 - 0.03614V^4 - 0.80737U^4V + \\ &0.90478U^3V^2 - 0.41391U^2V^3 + 0.63092UV^4 - \\ &0.22037V^5 - 0.13678U^6 + 0.69781U^5V - \\ &1.05556U^4V^2 + 1.32371U^3V^3 - 1.27196U^2V^4 + \\ &1.18766UV^5 - 0.12251V^6 + 1.05379U^6V - \\ &1.16127U^5V^2 + 2.61371U^4V^3 - 2.67357U^3V^4 - \\ &1.40762U^2V^5 + 0.34871UV^6 + 0.24145U^6V^2 - \\ &1.75051U^5V^3 + 3.73462U^4V^4 - 0.67849U^2V^6 - \\ &3.35164U^6V^3 + 2.09009U^5V^4 + 1.09491U^4V^5 + \\ &2.53857U^3V^6 + 0.68564U^6V^4 - 4.37226U^5V^5 + \\ &4.27159U^6V^5 - 5.63715U^5V^6 + 2.00221U^6V^6\end{aligned} \qquad \text{(H-29)}$$

$$\begin{aligned}\Delta\lambda('') = &-17.24754 - 0.13580U - 0.66515V - 0.02801U^2 - \\ &0.06547UV - 0.03005V^2 + 0.11463U^3 + \\ &0.04961U^2V + 0.04619UV^2 - 0.16246V^3 + \\ &0.17221U^4 + 0.34837U^3V + 0.82143U^2V^2 + \\ &0.82887UV^3 - 0.16253V^4 - 0.19168U^5 - \\ &0.23432U^4V - 0.99251U^3V^2 + 0.36556UV^4 + \\ &0.24159V^5 - 0.23927U^6 - 2.55220U^4V^2 - \\ &4.06721U^3V^3 - 2.08463U^2V^4 - 1.35955UV^5 + \\ &0.28261V^6 + 0.69600U^6V + 0.72875U^5V^2 + \\ &1.72449U^4V^3 + 0.53254U^2V^5 - 1.03470UV^6 + \\ &2.17999U^6V^2 + 3.24002U^5V^3 + 7.82174U^4V^4 + \\ &6.43003U^3V^5 + 1.80974U^2V^6 - 4.10731U^6V^3 + \\ &4.26679U^5V^4 - 6.74247U^4V^5 + 3.02834U^3V^6 - \\ &9.71198U^6V^4 - 11.53780U^4V^6 + 2.76442U^6V^5 - \\ &1.04831U^5V^6 - 8.03332U^6V^6\end{aligned} \qquad \text{(H-30)}$$

The North/South MREs with top power 3 (D48→D96) are as follows:

$$\Delta\phi('') = \begin{cases} -1.09217 + 0.13970V + 0.01241V^2 + 0.03306V^3 - \\ 0.25042U + 0.03551UV - 0.09889UV^2 - 0.25622UV^3 \\ -0.02857U^2 - 0.21053U^2V + 0.09449U^2V^2 + \\ 0.57587U^2V^3 + 0.02626U^3 + 0.11426U^3V - \\ 0.04984U^3V^2 - 0.31211U^3V^3 \end{cases} \text{if } U>0 \qquad \text{(H-31)}$$

$$\begin{cases} +0.0611U^2 + 0.36197U^2V - 0.40614U^2V^2 - \\ 1.12546U^2V^3 + 0.02703U^3 + 0.30818U^3V - \\ 0.27361U^3V^2 - 0.86592U^3V^3 \end{cases} \text{if } U\leq 0$$

$$\Delta\lambda('') = -17.24751 - 0.67457V - 0.04489V^2 - 0.02927V^3 - 0.14149U - 0.05497UV + 0.08902UV^2 + 0.24225UV^3 + 0.08509U^2 + 0.29721U^2V - 0.33338U^2V^2 - 0.76662U^2V^3 - 0.05142U^3 - 0.26469U^3V + 0.18455U^3V^2 + 0.66875U^3V^3 \quad \text{if } U>0$$

$$- 0.07004U^2 - 0.32481U^2V + 0.77069U^2V^2 + 1.60026U^2V^3 - 0.07456U^3 - 0.28755U^3V + 0.64704U^3V^2 + 1.39839U^3V^3 \quad \text{if } U\leq 0 \qquad \text{(H-32)}$$

The North/South MREs with top power 4 (D48→D96) are as follows:

$$\Delta\phi('') = -1.09120 + 0.13677V + 0.05685V^3 + 0.04060V^4 - 0.27175U + 0.15792UV + 0.10197UV^2 - 0.74695UV^3 - 0.60725UV^4 - 0.94999U^2V + 2.81137U^2V^3 + 1.39416U^2V^4 + 0.14599U^3 + 1.67119U^3V - 1.57520U^3V^2 - 4.01286U^3V^3 + 0.18882U^4 - 0.99026U^4V + 1.83261U^4V^2 + 1.80642U^4V^3 - 1.01923U^4V^4 \quad \text{if } U>0$$

$$- 0.14444U^2 + 1.15210U^2V + 1.68152U^2V^2 - 2.92383U^2V^3 - 3.32993U^2V^4 - 0.44885U^3 + 1.90224U^3V + 4.91998U^3V^2 - 2.41191U^3V^3 - 5.61039U^3V^4 - 0.31461U^4 + 0.98157U^4V + 3.62486U^4V^2 - 2.92708U^4V^4 \quad \text{if } U\leq 0 \qquad \text{(H-33)}$$

$$\Delta\lambda('') = -17.24685 - 0.67162V - 0.03552V^2 - 0.04874V^3 - 0.03196V^4 - 0.11429U - 0.07916UV - 0.31374UV^2 + 0.37119UV^3 + 0.64261UV^4 - 0.25104U^2 + 0.68360U^2V + 2.42906U^2V^2 - 1.55313U^2V^3 - 3.76435U^2V^4 + 0.91577U^3 - 2.10517U^3V - 4.80654U^3V^2 + 3.66179U^3V^3 + 5.93253U^3V^4 - 0.83006U^4 + 2.31270U^4V + 2.13105U^4V^2 - 3.58973U^4V^3 - 1.65111U^4V^4 \quad \text{if } U>0$$

$$+ 0.08076U^2 - 2.09753U^2V^2 - 3.10397U^2V^3 - 1.42310U^2V^4 + 0.21341U^3 + 0.75690U^3V - 5.64890U^3V^2 - 14.36463U^3V^3 - 9.12429U^3V^4 + 0.17314U^4 + 0.71614U^4V - 4.05899U^4V^2 - 11.51651U^4V^3 - 7.43560U^4V^4 \quad \text{if } U\leq 0 \qquad \text{(H-34)}$$

The East/West MREs with top power 3 (D48→D96) are as follows:

$$\Delta\phi('') = -1.09231 - 0.26115U + 0.01038U^2 - 0.01873U^3 + 0.12695V + 0.04653U^2V + 0.14143V^2 - 0.39282UV^2 - 0.17019U^2V^2 + 0.21431U^3V^2 - 0.18369V^3 + 0.55504UV^3 - 0.14056U^2V^3 - 0.09343U^3V^3 \quad \text{if } V>0$$

$$- 0.03802V^2 + 0.14489UV^2 - 0.05914U^2V^2 - 0.16570U^3V^2 - 0.01114V^3 + 0.12796UV^3 - 0.13784U^2V^3 - 0.21518U^3V^3 \quad \text{if } V\leq 0 \qquad \text{(H-35)}$$

$$\Delta\lambda('') = -17.24897 - 0.12003U + 0.00656U^2 - 0.01179U^3 - 0.67888V + 0.13498U^2V + 0.24708U^3V + 0.02408V^2 - 0.15778UV^2 - 0.89658U^3V^2 - 0.11263V^3 + 0.39806UV^3 - 0.66671U^2V^3 + 1.05906U^3V^3 \quad \text{if } V>0$$

$$- 0.03456V^2 - 0.16462UV^2 + 0.80556U^2V^2 + 1.24882U^3V^2 - 0.01130V^3 - 0.16335UV^3 + 0.91993U^2V^3 + 1.31849U^3V^3 \quad \text{if } V\leq 0 \qquad \text{(H-36)}$$

The East/West MREs with top power 4 (D48→D96) are as follows:

$$
\begin{aligned}
\Delta\phi('') = & -1.09410 - 0.25531U + 0.03175U^2 - 0.03465U^3 - \\
& 0.02987U^4 + 0.11637V + 0.15490U^2V - \\
& 0.08103U^3V - 0.29694U^4V \\
& + 0.25220V^2 - 0.42150UV^2 - 0.90288U^2V^2 + \\
& 0.90042U^3V^2 + 1.06799U^4V^2 - 0.46232V^3 + \\
& 0.66789UV^3 + 1.48213U^2V^3 - 2.70880U^3V^3 + \\
& 0.17423V^4 - 1.26175U^2V^4 + 2.34352U^3V^4 - \\
& 0.92872U^4V^4 \qquad \text{if } V>0 \\
& - 0.02756V^2 - 0.21028UV^2 - 0.47347U^2V^2 + \\
& 0.34970U^3V^2 + 0.06676V^3 - 0.73017UV^3 - \\
& 1.73571U^2V^3 + 0.70345U^3V^3 + 0.60302U^4V^3 + \\
& 0.06276V^4 - 0.49894UV^4 - 1.15535U^2V^4 \qquad \text{if } V\le 0
\end{aligned}
\tag{H-37}
$$

$$
\begin{aligned}
\Delta\lambda('') = & -17.24738 - 0.12965U - 0.02540U^2 + 0.02304U^3 + \\
& 0.06533U^4 - 0.61961V - 0.07180UV - \\
& 0.24467U^2V + 0.30735U^3V + 0.59776U^4V \\
& - 0.43179V^2 + 0.48582UV^2 + 2.96579U^2V^2 - \\
& 1.61620U^3V^2 - 5.25734U^4V^2 + 0.83985V^3 - \\
& 1.01512UV^3 - 5.97019U^2V^3 + 1.87775U^3V^3 + \\
& 10.48398U^4V^3 - 0.60947V^4 + 0.98484UV^4 + \\
& 2.68807U^2V^4 - 6.23149U^4V^4 \qquad \text{if } V>0 \\
& + 0.31092V^2 - 0.22303UV^2 - 0.24652U^2V^2 + \\
& 0.65249U^4V^2 + 0.62957V^3 - 2.13709U^3V^3 - \\
& 1.16528U^4V^3 + 0.36903V^4 + 0.14998UV^4 - \\
& 0.16957U^2V^4 - 2.17783U^3V^4 - 1.32952U^4V^4 \qquad \text{if } V\le 0
\end{aligned}
\tag{H-38}
$$

The Four-Quadrant MREs with top power 3 (D48→D96) are as follows:

$$
\begin{aligned}
\Delta\phi('') = & -1.09253 - 0.25830U + 0.13910V + 0.01245U^2 + \\
& 0.04166UV + 0.01013V^2 - 0.02591U^3 - \\
& 0.06201UV^2 + 0.03073V^3 - 0.12029U^3V - \\
& 0.26082UV^3 \\
& - 0.80606U^2V^2 + 0.99655U^3V^2 + 1.23023U^2V^3 - 1.1167U^3V^3 \qquad \text{if } U>0 \ \&\ V>0 \\
& + 0.97722U^2V^2 - 1.31629U^3V^2 + 1.57382U^2V^3 - 1.81679U^3V^3 \qquad \text{if } U>0 \ \&\ V\le 0 \\
& + 4.71460U^2V^2 + 6.60863U^3V^2 - 11.74656U^2V^3 - 17.37754U^3V^3 \qquad \text{if } U\le 0 \ \&\ V>0 \\
& - 1.43853U^2V^2 - 1.47484U^3V^2 - 2.04927U^2V^3 - 1.88651U^3V^3 \qquad \text{if } U\le 0 \ \&\ V\le 0
\end{aligned}
\tag{H-39}
$$

$$
\begin{aligned}
\Delta\lambda('') = & -17.24800 - 0.12262U - 0.67480V + 0.00281U^2 - \\
& 0.02055UV - 0.03261V^2 - 0.00396U^3 + \\
& 0.11967U^2V - 0.01573UV^2 - 0.01565V^3 + \\
& 0.31449U^3V + 0.10103UV^3 \\
& + 0.16846U^2V^2 - 1.29146U^3V^2 - 0.61762U^2V^3 + 1.32457U^3V^3 \qquad \text{if } U>0 \ \&\ V>0 \\
& - 0.29505U^2V^2 + 2.79154U^3V^2 - 0.57246U^2V^3 + 3.18561U^3V^3 \qquad \text{if } U>0 \ \&\ V\le 0 \\
& - 1.80338U^2V^2 - 5.05514U^3V^2 + 5.48569U^2V^3 + 15.25399U^3V^3 \qquad \text{if } U\le 0 \ \&\ V>0 \\
& + 1.49241U^2V^2 + 1.95359U^3V^2 + 1.99797U^2V^3 + 2.27176U^3V^3 \qquad \text{if } U\le 0 \ \&\ V\le 0
\end{aligned}
\tag{H-40}
$$

The Four-Quadrant MREs with top power 4 (D48→D96) are as follows:

$$
\begin{aligned}
\Delta\phi('') = & -1.09268 - 0.25768U + 0.13639V + 0.01277U^2 + \\
& 0.04207UV + 0.01381V^2 - 0.02734U^3 - \\
& 0.02898U^2V - 0.07497UV^2 + 0.06332V^3 +
\end{aligned}
\tag{H-41}
$$

$$
\begin{cases}
0.00493U^4 - 0.16625U^3V - 0.53312UV^3 + \\
0.02831V^4 - 0.34255UV^4 \\
- 0.45449U^2V^2 + 1.41577U^3V^2 + 1.62556U^2V^3 - \\
0.66050U^4V^2 - 3.86133U^3V^3 + 1.25318U^2V^4 + \\
2.74044U^4V^3 - 1.12916U^4V^4
\end{cases} \quad \text{if } U>0 \ \& \ V>0
$$

$$
\begin{cases}
+ 3.44245U^2V^2 - 22.36761U^3V^2 + 9.51474U^2V^3 + \\
34.39782U^4V^2 - 60.62440U^3V^3 + 6.48778U^2V^4 + \\
94.78947U^4V^3 - 42.48863U^3V^4 + 66.64619U^4V^4
\end{cases} \quad \text{if } U>0 \ \& \ V\leq 0
$$

$$
\begin{cases}
- 2.65391U^2V^2 - 22.56437U^3V^2 + 41.70016U^2V^3 - \\
27.52312U^4V^2 + 205.65453U^3V^3 - 99.10100U^2V^4 + \\
220.33167U^4V^3 - 419.12503U^3V^4 - 436.84328U^4V^4
\end{cases} \quad \text{if } U\leq 0 \ \& \ V>0
$$

$$
\begin{cases}
- 1.12562U^2V^2 - 1.58546U^2V^3 + 1.06190U^4V^2 + \\
2.72768U^3V^3 + 3.69863U^4V^3 + 3.43845U^3V^4 + \\
3.03975U^4V^4
\end{cases} \quad \text{if } U\leq 0 \ \& \ V\leq 0
$$

$$
\begin{aligned}
\Delta\lambda('') = & -17.24829 - 0.12822U - 0.67291V - 0.01197U^2 - \\
& 0.05692UV - 0.03373V^2 + 0.01329U^3 + \\
& 0.29698U^2V - 0.14875UV^2 - 0.05910V^3 + \\
& 0.03763U^4 + 0.24099U^3V + 0.43139UV^3 - \\
& 0.04775V^4 - 0.17435U^4V + 0.55425UV^4
\end{aligned} \tag{H-42}
$$

$$
\begin{cases}
+ 3.91356U^2V^2 - 11.41675U^3V^2 - 9.07714U^2V^3 + \\
6.82614U^4V^2 + 23.99470U^3V^3 + 2.75649U^2V^4 - \\
14.97517U^4V^3 - 10.64716U^3V^4 + 7.61079U^4V^4
\end{cases} \quad \text{if } U>0 \ \& \ V>0
$$

$$
\begin{cases}
+ 12.72639U^2V^2 - 54.82024U^3V^2 + 36.09085U^2V^3 + \\
73.02009U^4V^2 - 175.13081U^3V^3 + 25.32003U^2V^4 + \\
234.06656U^4V^3 - 131.08910U^3V^4 + 176.69355U^4V^4
\end{cases} \quad \text{if } U>0 \ \& \ V\leq 0
$$

$$
\begin{cases}
+ 4.59944U^2V^2 + 29.55962U^3V^2 - 36.80249U^2V^3 + \\
28.91412U^4V^2 - 190.16377U^3V^3 + 55.43907U^2V^4 - \\
177.73922U^4V^3 + 270.89916U^3V^4 + 253.48171U^4V^4
\end{cases} \quad \text{if } U\leq 0 \ \& \ V>0
$$

$$
\begin{cases}
+ 2.89976U^2V^2 - 0.48261U^3V^2 + 7.75994U^2V^3 - \\
3.84808U^4V^2 - 5.13270U^2V^4 - 7.97219U^4V^3 - \\
4.96697U^4V^4
\end{cases} \quad \text{if } U\leq 0 \ \& \ V\leq 0
$$

The Chebyshev MREs with top power 3 (D48→D96) are as follows:

$$
\begin{aligned}
\Delta\phi('') = & -1.09787 - 0.27319T_1(U) + 0.14339T_1(V) \\
& - 0.00705T_2(U) - 0.0162T_1(U)T_1(V) \\
& - 0.01206T_2(V) - 0.00914T_2(U)T_1(V) \\
& + 0.01253T_3(U)T_1(V) - 0.01240T_2(U)T_2(V) \\
& + 0.01069T_1(U)T_3(V) + 0.00375T_3(U)T_2(V) \\
& - 0.00287T_2(U)T_3(V) + 0.00425T_3(U)T_3(V)
\end{aligned} \tag{H-43}
$$

$$
\begin{aligned}
\Delta\lambda('') = & -17.25260 - 0.16766T_1(U) - 0.64453T_1(V) \\
& + 0.00760T_2(U) - 0.01461T_2(V) \\
& - 0.00530T_3(U) + 0.03047T_2(U)T_1(V) \\
& - 0.05018T_1(U)T_2(V) + 0.01034T_3(V) \\
& + 0.01240T_3(U)T_1(V) - 0.00341T_2(U)T_2(V) \\
& - 0.00335T_1(U)T_3(V) - 0.00635T_3(U)T_2(V) \\
& + 0.00989T_2(U)T_3(V) + 0.00670T_3(U)T_3(V)
\end{aligned} \tag{H-44}
$$

The Chebyshev MREs with top power 4 (D48→D96) are as follows:

$$
\begin{aligned}
\Delta\phi('') = & -1.08683 - 0.31557T_1(U) + 0.18624T_1(V) - 0.00559T_2(U) - \\
& 0.05265T_1(U)T_1(V) - 0.00939T_3(U) + 0.03461T_2(U)T_1(V) - \\
& 0.05469T_1(U)T_2(V) + 0.02041T_3(V) - 0.00889T_4(U) + \\
& 0.01568T_3(U)T_1(V) - 0.01477T_2(U)T_2(V) + \\
& 0.00487T_1(U)T_3(V) - 0.00260T_4(V) + 0.00734T_4(U)T_1(V) - \\
& 0.00829T_3(U)T_2(V) + 0.02078T_2(U)T_3(V) - \\
& 0.01624T_1(U)T_4(V) - 0.01393T_4(U)T_2(V) + \\
& 0.01005T_3(U)T_3(V) - 0.00986T_2(U)T_4(V) +
\end{aligned} \tag{H-45}
$$

$$0.00678T_4(U)T_3(V) - 0.00499T_3(U)T_4(V) - 0.00882T_4(U)T_4(V)$$

$$\begin{aligned}\Delta\lambda('') = -17.21579 &- 0.18537T_1(U) - 0.63563T_1(V) + 0.08709T_2(U) - 0.12017T_1(U)T_1(V) + 0.03820T_2(V) - \\ &0.03006T_3(U) + 0.08361T_2(U)T_1(V) - 0.06034T_1(U)T_2(V) + 0.00235T_3(V) + 0.02593T_4(U) - 0.07021T_3(U)T_1(V) + \\ &0.10804T_2(U)T_2 - 0.06692T_1(U)T_3(V) + 0.01947T_4(V) + 0.01212T_4(U)T_1(V) - 0.03434T_3(U)T_2(V) + \\ &0.01448T_2(U)T_3(V) + 0.01137T_1(U)T_4(V) + 0.03568T_4(U)T_2(V) - 0.03337T_3(U)T_3(V) + \\ &0.03900T_2(U)T_4(V) - 0.00061T_4(U)T_3(V) + 0.01468T_4(U)T_4(V)\end{aligned} \quad \text{(H-46)}$$

The Chebyshev MREs with top power 5 (D48→D96) are as follows:

$$\begin{aligned}\Delta\phi('') = -1.00196 &- 0.50381T_1(U) + 0.36925T_1(V) + 0.14729T_2(U) - 0.35240T_1(U)T_1(V) + 0.09774T_2(V) - 0.11275T_3(U) + \\ &0.34820T_2(U)T_1(V) - 0.33908T_1(U)T_2(V) + 0.11029T_3(V) + 0.03387T_4(U) - 0.16752T_3(U)T_1(V) + 0.17016T_2(U)T_2(V) - \\ &0.10920T_1(U)T_3(V) + 0.02133T_4(V) - 0.00545T_5(U) + 0.08327T_4(U)T_1(V) - 0.16490T_3(U)T_2(V) + \\ &0.17592T_2(U)T_3(V) - 0.09013T_1(U)T_4(V) + 0.00990T_5(V) - 0.01177T_5(U)T_1(V) + 0.03579T_4(U)T_2(V) - \\ &0.06548T_3(U)T_3(V) + 0.04089T_2(U)T_4(V) - 0.01743T_1(U)T_5(V) - 0.01023T_5(U)T_2(V) + \\ &0.04140T_4(U)T_3(V) - 0.04363T_3(U)T_4(V) + 0.01892T_2(U)T_5(V) - 0.00284T_5(U)T_3(V) + \\ &0.00593T_4(U)T_4(V) - 0.01371T_3(U)T_5(V) + 0.00106T_4(U)T_5(V)\end{aligned} \quad \text{(H-47)}$$

$$\begin{aligned}\Delta\lambda('') = -17.34304 &+ 0.05813T_1(U) - 0.91519T_1(V) + 0.21241T_1(U)T_1(V) - 0.13281T_2(V) - 0.14024T_2(U)T_1(V) + \\ &0.26479T_1(U)T_2(V) - 0.11572T_3(V) + 0.04805T_4(U) - 0.11472T_3(U)T_1(V) + 0.07994T_1(U)T_3(V) - 0.02949T_4(V) - \\ &0.00362T_5(U) + 0.02932T_4(U)T_1(V) - 0.06554T_2(U)T_3(V) + 0.08331T_1(U)T_4(V) - 0.01793T_5(V) - 0.04549T_5(U)T_1(V) + \\ &0.07397T_4(U)T_2(V) - 0.07085T_3(U)T_3(V) + 0.01197T_2(U)T_4(V) + 0.00672T_1(U)T_5(V) - \\ &0.005032T_5(U)T_2(V) + 0.01769T_4(U)T_3(V) - 0.00298T_3(U)T_4(V) - 0.00889T_2(U)T_5(V) - \\ &0.02868T_5(U)T_3(V) + 0.02939T_4(U)T_4(V) - 0.0226T_3(U)T_5(V) - 0.00370T_5(U)T_4(V) + \\ &0.00379T_4(U)T_5(V) - 0.01187T_5(U)T_5(V)\end{aligned} \quad \text{(H-48)}$$

The Chebyshev MREs with top power 6 (D48→D96) are as follows:

$$\begin{aligned}\Delta\phi('') = -0.77283 &- 0.83279T_1(U) + 0.62993T_1(V) + 0.62075T_2(U) - 1.23980T_1(U)T_1(V) + 0.46777T_2(V) - 0.43660T_3(U) + \\ &0.97863T_2(U)T_1(V) - 0.75495T_1(U)T_2(V) + 021249T_3(V) + 0.28768T_4(U) - 0.90470T_3(U)T_1(V) + 0.92160T_2(U)T_2(V) - \\ &0.62293T_1(U)T_3(V) + 0.014748T_4(V) - 0.09915T_5(U) + 0.46084T_4(U)T_1(V) - 0.60560T_3(U)T_2(V) + 0.45880T_2(U)T_3(V) - \\ &0.24834T_1(U)T_4(V) + 0.05257T_5(V) + 0.02745T_6(U) - 0.21166T_5(U)T_1(V) + 0.43409T_4(U)T_2(V) - 0.48294T_3(U)T_3(V) + \\ &0.30526T_2(U)T_4(V) - 0.09456T_1(U)T_5(V) + 0.00469T_6(V) + 0.03901T_6(U)T_1(V) - 0.13832T_5(U)T_2(V) + 0.22007T_4(U)T_3(V) - \\ &0.21567T_3(U)T_4(V) + 0.11765T_2(U)T_5(V) - 0.04075T_1(U)T_6(V) + 0.04473T_6(U)T_2(V) - 0.11443T_5(U)T_3(V) + 0.14559T_4(U)T_4(V) - \\ &0.08471T_3(U)T_5(V) + 0.01813T_2(U)T_6(V) + 0.01622T_6(U)T_3(V) - 0.05125T_5(U)T_4(V) + 0.06038T_4(U)T_5(V) - 0.03603T_3(U)T_6(V) +\end{aligned} \quad \text{(H-49)}$$

$$0.01460T_6(U)T_4(V) - 0.01718T_5(U)T_5(V) + 0.01173T_4(U)T_6(V) + 0.00874T_6(U)T_5(V) - 0.01133T_5(U)T_6(V) + 0.00198T_6(U)T_6(V)$$

$$\begin{aligned}\Delta\lambda('') = {} & -17.89429 + 1.04086T_1(U) - 1.84321T_1(V) - 0.92760T_2(U) + \\ & 2.17642T_1(U)T_1(V) - 0.87373T_2(V) + 0.61024T_3(U) - \\ & 1.72516T_2(U)T_1(V) + 1.70050T_1(U)T_2(V) - 0.57560T_3(V) - \\ & 0.27627T_4(U) + 1.08482T_3(U)T_1(V) - 1.25867T_2(U)T_2(V) + \\ & 1.00278T_1(U)T_3(V) - 0.26876T_4(V) + 0.09642T_5(U) - \\ & 0.53905T_4(U)T_1(V) + 0.89253T_3(U)T_2(V) - 0.86190T_2(U)T_3(V) + \\ & 0.51418T_1(U)T_4(V) - 0.07881T_5(V) - 0.00876T_6(U) + \\ & 0.15292T_5(U)T_1(V) - 0.34776T_4(U)T_2(V) + 0.49454T_3(U)T_3(V) - \\ & 0.40305T_2(U)T_4(V) + 0.20625T_1(U)T_5(V) - 0.01779T_6(V) + \\ & 0.02287T_6(U)T_1(V) + 0.13735T_5(U)T_2(V) - 0.25864T_4(U)T_3(V) + \\ & 0.26964T_3(U)T_4(V) - 0.12549T_2(U)T_5(V) + 0.02413T_1(U)T_6(V) + \\ & 0.05079T_5(U)T_3(V) - 0.09276T_4(U)T_4(V) + 0.09754T_3(U)T_5(V) - \\ & 0.03229T_2(U)T_6(V) - 0.00743T_6(U)T_3(V) + 0.02897T_5(U)T_4(V) - \\ & 0.02752T_4(U)T_5(V) + 0.01852T_3(U)T_6(V) + 0.00918T_6(U)T_4(V) + \\ & 0.00182T_4(U)T_6(V) + 0.00448T_6(U)T_5(V) + 0.00782T_6(U)T_6(V)\end{aligned} \tag{H-50}$$

The MREs of Ord6tr(Lat) and EW4tr(Lon) (D48→D96) is as follows.

$$\begin{aligned}\Delta\phi('') = {} & -1.09184 - 0.25923U + 0.13427V + 0.01258V^2 - \\ & 0.02234U^3 + 0.11929U^2V - 0.13431UV^2 + \\ & 0.09536V^3 + 0.05600U^4 - 0.11519U^3V - \\ & 0.28478UV^3 - 0.25322U^4V + 0.29737U^3V^2 - \\ & 0.52256U^2V^3 + 0.44093UV^4 - 0.17377V^5 - \\ & 0.05652U^6 + 0.37998U^3V^3 - 0.39517U^2V^4 + \\ & 0.58467UV^5 - 0.11294V^6 + 1.33082U^4V^3 - \\ & 1.35697U^3V^4 - 0.33803U^5V^3 + 0.75580U^4V^4 - \\ & 0.50114U^6V^3 + 1.04331U^3V^6 + 1.01810U^6V^4 - \\ & 1.65939U^5V^5 + 1.22406U^6V^5 - 1.63569U^5V^6\end{aligned} \tag{H-51}$$

$$\begin{aligned}\Delta\lambda('') = {} & -17.24931 - 0.13101U - 0.01532U^2 + \\ & 0.02545U^3 + 0.05804U^4 - 0.62942V - \\ & 0.12471U^2V + 0.11838U^3V + 0.32602U^4V \\ & - 0.18680V^2 + 0.08986UV^2 + 1.41112U^2V^2 - \\ & 0.55717U^3V^2 - 2.9331U^4V^2 + 0.07985V^3 - \\ & 2.48610U^2V^3 + 5.56784U^4V^3 + 1.23345U^2V^4 - \\ & 2.65893U^4V^4 \qquad \text{if } V>0 \\ & + 0.30075V^2 - 0.03472UV^2 + 0.40479U^4V^2 + \\ & 0.64266V^3 - 0.68551U^3V^3 + 0.38714V^4 - \\ & 0.34671U^2V^4 - 0.74364U^3V^4 \qquad \text{if } V\le 0\end{aligned} \tag{H-52}$$

H.3 Great Britain (ETRS89 to OSGB36)

The normalised intermediate coordinates U and V were defined as follows:

$$U = 0.14662\left(\phi_{\text{in deg}} - 56.104\right); \tag{H-53}$$

$$V = 0.12769\left(\lambda_{\text{in deg}} + 2.968\right). \tag{H-54}$$

The Ordinary MREs with top power 3 (ETRS89→OSGB36) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & 0.26880 + 3.13019U - 0.48925V + 0.03480U^2 - \\ & 0.13513UV - 0.12063V^2 - 0.41272U^3 + 0.04574U^2V - \\ & 0.05737UV^2 - 0.01779V^3 - 0.13797U^3V - \\ & 0.10340U^2V^2 - 0.04485UV^3 + 0.28145U^3V^2 - \\ & 0.03390U^2V^3 + 0.26330U^3V^3\end{aligned} \tag{H-55}$$

$$\begin{aligned}\Delta\lambda('') = {} & 5.26539 + 1.13493U + 3.84616V + 0.29140U^2 + \\ & 0.96264UV - 0.13459V^2 - 0.67029U^2V + \\ & 0.32858UV^2 - 0.18865V^3 - 0.14764U^3V + \\ & 0.09906U^2V^2 - 0.56623UV^3 - 0.54143U^3V^2 + \\ & 0.73524U^2V^3 + 0.51748U^3V^3\end{aligned} \quad \text{(H-56)}$$

The Ordinary MREs with top power 4 (ETRS89→OSGB36) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & 0.27694 + 3.10212U - 0.49083V - 0.09106U^2 - \\ & 0.17353UV - 0.33563V^2 - 0.38997U^3 + 0.03572U^2V + \\ & 0.37465UV^2 - 0.06037V^3 + 0.16810U^4 - 0.07980U^3V + \\ & 1.77928U^2V^2 + 0.35374V^4 + 0.03758U^4V + 0.20333U^3V^2 + \\ & 0.34716U^2V^3 - 0.52576UV^4 - 1.92044U^4V^2 + \\ & 0.24741U^3V^3 - 2.50028U^2V^4 - 0.47975U^4V^3 + \\ & 2.57554U^4V^4\end{aligned} \quad \text{(H-57)}$$

$$\begin{aligned}\Delta\lambda('') = {} & 5.26440 + 1.12771U + 3.96188V + 0.28801U^2 + \\ & 0.67612UV - 0.09858V^2 + 0.01050U^3 - 2.31902U^2V + \\ & 1.15050UV^2 - 0.18429V^3 + 0.08247U^3V + \\ & 2.03877U^2V^2 - 0.07408UV^3 - 0.09743V^4 + \\ & 2.10700U^4V - 1.50041U^3V^2 + 1.65599U^2V^3 - \\ & 1.42274UV^4 - 2.00055U^4V^2 + 0.24651U^3V^3 - \\ & 1.65350U^2V^4 - 1.67385U^4V^3 + 1.72665U^3V^4 + \\ & 1.68103U^4V^4\end{aligned} \quad \text{(H-58)}$$

The Ordinary MREs with top power 5 (ETRS89→OSGB36) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & 0.28523 + 3.18129U - 0.37706V - 0.21549U^2 + \\ & 0.02293UV - 0.07328V^2 - 0.98747U^3 - \\ & 0.13604U^2V + 1.00600UV^2 - 0.43526V^3 + \\ & 025407U^4 - 0.56438U^3V + 0.86061U^2V^2 + \\ & 0.89751UV^3 - 0.04742V^4 + 0.65050U^5 - \\ & 0.94508U^3V^2 + 3.61119U^2V^3 - 1.45349UV^4 + \\ & 0.47929V^5 + 0.43736U^5V - 1.17595U^4V^2 - \\ & 1.61289U^3V^3 - 0.74342U^2V^4 - 2.12472UV^5 + \\ & 0.16123U^5V^2 - 3.39180U^4V^3 + 2.16065U^3V^4 - \\ & 5.20199U^2V^5 + 0.88148U^5V^3 + 1.09940U^4V^4 + \\ & 6.97010U^3V^5 - 0.96981U^5V^4 + 5.20614U^4V^5 - \\ & 5.37890U^5V^5\end{aligned} \quad \text{(H-59)}$$

$$\begin{aligned}\Delta\lambda('') = {} & 5.25889 + 1.10735U + 3.92559V + 0.30799U^2 + \\ & 0.54573UV + 0.17112V^2 + 0.11572U^3 - \\ & 2.08435U^2V + 1.84957UV^2 + 0.72085V^3 - \\ & 0.01266U^4 + 0.93227U^3V + 1.31930U^2V^2 + \\ & 2.12789UV^3 - 0.38721V^4 - 0.10981U^5 + \\ & 1.99680U^4V - 4.37713U^3V^2 - 3.15828U^2V^3 - \\ & 2.19265UV^4 - 1.17070V^5 - 0.87501U^5V - \\ & 1.70121U^4V^2 - 11.03630U^3V^3 - 1.05280U^2V^4 - \\ & 2.61099UV^5 + 2.39468U^5V^2 + 1.31556U^4V^3 + \\ & 4.97593U^3V^4 + 5.69917U^2V^5 + 10.49728U^5V^3 + \\ & 1.55887U^4V^4 + 12.51239U^3V^5 - 2.70000U^5V^4 - \\ & 3.02424U^4V^5 - 12.00659U^5V^5\end{aligned} \quad \text{(H-60)}$$

The Ordinary MREs with top power 6 (ETRS89→OSGB36) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & 0.28527 + 3.15324U - 0.36119V - 0.26092U^2 - \\ & 0.01776V^2 - 0.75015U^3 - 0.48132U^2V + 1.0487UV^2 - \\ & 0.97505V^3 + 0.67441U^4 - 0.67991U^3V - 0.85958U^2V^2 - \\ & 1.34678UV^3 - -1.60426V^4 + 0.46304U^5 + 0.55127U^4V - \\ & 4.34242U^3V^2 + 8.27329U^2V^3 - 2.07352UV^4 + \\ & 1.17088V^5 - 0.40986U^6 + 0.62947U^5V - 0.97542U^4V^2 + \\ & 8.55764U^3V^3 + 10.63596U^2V^4 + 0.98671UV^5 + \\ & 2.10508V^6 - 0.23723U^6V + 3.41112U^5V^2 - \\ & 5.06445U^4V^3 + 19.40149U^3V^4 - 12.50879U^2V^5 + \\ & 1.78234U^6V^2 - 7.45374U^5V^3 - 9.50118U^4V^4 - \\ & 4.80034U^3V^5 - 11.75161U^2V^6 - 2.46012U^6V^3 - \\ & 17.46097U^5V^4 + 11.72029U^4V^5 - 15.69240U^3V^6 + \\ & 3.76650U^5V^5 + 15.20530U^4V^6 + 15.58256U^5V^6 - \\ & 4.97200U^6V^6\end{aligned} \quad \text{(H-61)}$$

$$\begin{aligned}\Delta\lambda('') = {} & 5.26318 + 1.10222U + 3.89569V + 0.21601U^2 + \\ & 0.22377UV - 0.25139V^2 + 0.15728U^3 - 3.01752U^2V + \\ & 0.35922V^3 + 0.33956U^4 + 2.60618U^3V + 2.84976U^2V^2 - \\ & 1.97404UV^3 + 0.57646V^4 - 0.129840U^5 + 6.89722U^4V - \\ & 3.85766U^2V^3 + 1.05103UV^4 - 0.52422V^5 - 0.29441U^6 - \\ & 1.90081U^5V - 6.87715U^4V^2 + 4.77763U^3V^3 - \\ & 19.18008U^2V^4 + 3.32443UV^5 - 0.56100V^6 - \\ & 4.22160U^6V - 0.57332U^5V^2 + 5.81960U^4V^3 - \\ & 21.17460U^3V^4 + 4.76352U^2V^5 - 1.72683UV^6 + \\ & 4.53718U^6V^2 - 3.30271U^5V^3 + 38.01589U^4V^4 - \\ & 11.17777U^3V^5 + 23.07252U^2V^6 - 2.42298U^6V^3 + \\ & 23.32072U^5V^4 - 5.97112U^4V^5 + 27.26517U^3V^6 - \\ & 21.31544U^6V^4 + 8.40572U^5V^5 - 50.86119U^4V^6 + \\ & 1.64768U^6V^5 - 29.62809U^5V^6 + 31.44880U^6V^6\end{aligned} \quad \text{(H-62)}$$

The North/South MREs with top power 3 (ETRS89→OSGB36) are as follows:

$$\begin{aligned}\Delta\phi('') = {} & 0.28538 - 0.41802V - 0.11961V^2 - 0.10169V^3 + \\ & 3.28408U + 0.27681UV + 0.57604UV^2 - 1.2314UV^3 \\ & - 0.098424U^2 - 1.45485U^2V - 1.52211U^2V^2 + \\ & 3.95979U^2V^3 + 0.57734U^3 + 0.98805U^3V + \\ & 0.95184U^3V^2 - 2.65510U^3V^3 \\ & + 0.37799U^2 + 0.87948U^2V + 1.71478U^2V^2 - \\ & 3.07516U^2V^3 - 0.19022U^3 + 0.42040U^3V + \\ & 1.51362U^3V^2 - 1.80253U^3V^3\end{aligned} \quad \begin{array}{l}\text{if } U>0 \\ \\ \\ \\ \text{if } U\leq 0\end{array} \quad \text{(H-63)}$$

$$\begin{aligned}\Delta\lambda('') = {} & 5.26013 + 3.90932V - 0.18460V^2 - 0.07808V^3 + \\ & 1.06886U + 0.61534UV^2 \\ & + 0.64301U^2 - 0.84033U^2V - 0.94478U^2V^2 + \\ & 1.05226U^2V^3 - 0.30653U^3 + 1.40503U^3V + \\ & 0.33394U^3V^2 - 0.98447U^3V^3 \\ & + 0.09984U^2 - 5.12664U^2V + 2.85239U^2V^2 + \\ & 1.76486U^2V^3 - 0.15003U^3 - 4.20115U^3V + \\ & 1.94654U^3V^2 + 1.97842U^3V^3\end{aligned} \quad \begin{array}{l}\text{if } U>0 \\ \\ \\ \\ \text{if } U\leq 0\end{array} \quad \text{(H-64)}$$

The North/South MREs with top power 4 (ETRS89→OSGB36) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.28463 - 0.38989V + 0.03226V^2 - 0.05433V^3 - \\
& 0.21613V^4 + 3.09622U - 0.14121UV + \\
& 2.02403UV^2 - 0.61875UV^3 - 2.37658UV^4 \\
& + 0.28590U^2 + 0.83498U^2V - 13.56386U^2V^2 - \\
& 1.62511U^2V^3 + 15.78044U^2V^4 - 2.02956U^3 - \\
& 2.71365U^3V + 28.94118U^3V^2 + 7.87872U^3V^3 - \\
& 32.59947U^3V^4 + 1.52979U^4 + 1.82684U^4V - \\
& 17.70553U^4V^2 - 5.68878U^4V^3 + 19.60039U^4V^4 \qquad \text{if } U>0 \\
& -0.83308U^2 - 1.02488U^2V + 1.66396U^2V^2 - \\
& 3.69544U^2V^3 + 5.04643U^2V^4 - 2.55853U^3 - \\
& 2.56530U^3V - 4.76703U^3V^2 - 5.68856U^3V^3 + \\
& 24.25362U^3V^4 - 1.43792U^4 - 1.73773U^4V - \\
& 5.29643U^4V^2 - 2.46501U^4V^3 + 18.26279U^4V^4 \qquad \text{if } U\leq 0
\end{aligned}
\tag{H-65}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.26576 + 3.92025V - 0.22452V^2 + 0.01086V^3 + \\
& 0.15087V^4 + 1.14765U - 0.60366UV - 0.75303UV^2 \\
& - 0.54267U^2 + 0.41385U^2V + 4.19007U^2V^2 + \\
& 2.51835U^3 + 2.48937U^3V^3 - 8.09092U^3V^4 - \\
& 1.73444U^4 + 0.76159U^4V - 3.52937U^4V^2 - \\
& 2.58940U^4V^3 + 8.14742U^4V^4 \qquad \text{if } U>0 \\
& + 0.30674U^2 - 9.21483U^2V + 0.61878U^2V^2 - \\
& 2.00977U^2V^3 - 9.46806U^2V^4 - 11.97644U^3V + \\
& 2.74386U^3V^2 - 6.79974U^3V^3 - 22.79699U^3V^4 - \\
& 4.57198U^4V + 1.77202U^4V^2 - 4.72678U^4V^3 - \\
& 13.82478U^4V^4 \qquad \text{if } U\leq 0
\end{aligned}
\tag{H-66}
$$

The East/West MREs with top power 3 (ETRS89→OSGB36) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.27744 + 3.13599U + 0.00939U^2 - 0.43943U^3 - \\
& 0.32189V + 0.09416UV - 0.21753U^2V - \\
& 0.34607U^3V \\
& -1.63847V^2 - 2.14958UV^2 + 2.25304U^2V^2 + \\
& 2.91722U^3V^2 + 1.54383V^3 + 1.83015UV^3 - \\
& 2.49052U^2V^3 - 2.08492U^3V^3 \qquad \text{if } V>0 \\
& + 0.34700V^2 + 1.24338UV^2 - 0.68258U^2V^2 - \\
& 0.50209U^3V^2 + 0.30559V^3 + 1.30297UV^3 - \\
& 0.49754U^2V^3 - 0.51048U^3V^3 \qquad \text{if } V\leq 0
\end{aligned}
\tag{H-67}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.25919 + 1.10428U + 0.28462U^2 + 0.02591U^3 + \\
& 3.79927V + 0.63780UV - 1.47893U^2V - \\
& 0.46291U^3V \\
& + 1.12017V^2 + 4.86019UV^2 + 2.37158U^2V^2 - \\
& 3.89781U^3V^2 - 1.57977V^3 - 5.31459UV^3 - \\
& 0.89247U^2V^3 + 5.08622U^3V^3 \qquad \text{if } V>0 \\
& - 0.34600V^2 + 0.57439UV^2 - 2.45204U^2V^2 - \\
& 2.59009U^3V^2 - 0.41719V^3 + 0.37867UV^3 - \\
& 1.19413U^2V^3 - 1.47762U^3V^3 \qquad \text{if } V\leq 0
\end{aligned}
\tag{H-68}
$$

The East/West MREs with top power 4 (ETRS89→OSGB36) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.27824 + 3.10630U - 0.07367U^2 - 0.39461U^3 + \\
& 0.14801U^4 - 0.38988V + 0.16474UV - \\
& 1.23256U^2V - 0.94560U^3V + 1.35619U^4V \\
& \left.\begin{aligned}
& + 0.64396V^2 + 1.16381UV^2 + 9.22456U^2V^2 + \\
& 3.14371U^3V^2 - 11.56304U^4V^2 - 8.89897V^3 - \\
& 15.98022UV^3 - 11.29031U^2V^3 + 9.96900U^3V^3 + \\
& 22.61363U^4V^3 + 9.37244V^4 + 15.30214UV^4 - \\
& 12.84731U^3V^4 - 10.47660U^4V^4
\end{aligned}\right\} \quad \text{if } V>0 \\
& \left.\begin{aligned}
& - 0.25492V^2 + 4.23624UV^2 - 8.32153U^2V^2 - \\
& 7.29840U^3V^2 + 10.98840U^4V^2 - 1.48419V^3 + \\
& 13.19812UV^3 - 20.08544U^2V^3 - 21.66802U^3V^3 + \\
& 30.32125U^4V^3 - 1.46828V^4 + 11.10127UV^4 - \\
& 15.23526U^2V^4 - 18.30653U^3V^4 + 24.48623U^4V^4
\end{aligned}\right\} \quad \text{if } V\leq 0
\end{aligned}
\tag{H-69}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.25967 + 1.11225U + 0.26827U^2 + 0.02009U^3 + \\
& 0.02169U^4 + 3.82693V + 0.43770UV - \\
& 3.13103U^2V + 0.29353U^3V + 3.90899U^4V \\
& \left.\begin{aligned}
& + 1.56386V^2 + 3.41717UV^2 + 7.31814U^2V^2 - \\
& 3.14290U^3V^2 - 14.49299U^4V^2 - 3.87954V^3 - \\
& 2.18150UV^3 - 6.95256U^2V^3 + 21.24362U^4V^3 + \\
& 2.18452V^4 - 1.97341UV^4 + 3.03729U^2V^4 + \\
& 4.41537U^3V^4 - 11.61360U^4V^4
\end{aligned}\right\} \quad \text{if } V>0 \\
& \left.\begin{aligned}
& - 0.61398V^2 + 0.76087UV^2 - 3.10913U^2V^2 - \\
& 2.43862U^3V^2 + 9.43313U^4V^2 - 0.56560V^3 - \\
& 8.73433U^2V^3 - 4.75519U^3V^3 + 21.09019U^4V^3 - \\
& 1.04396UV^4 - 8.08129U^2V^4 - 3.05040U^3V^4 + \\
& 15.92383U^4V^4
\end{aligned}\right\} \quad \text{if } V\leq 0
\end{aligned}
\tag{H-70}
$$

The Four-Quadrant MREs with top power 3 (ETRS89→OSGB36) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.27443 + 3.13616U - 0.45338V + 0.03590U^2 + \\
& 0.14722UV - 0.10998V^2 - 0.41639U^3 + \\
& 0.12004U^2V + 1.03617UV^2 - 0.02489V^3 - \\
& 0.28332U^3V - 1.06231UV^3 \\
& \left.\begin{aligned}
& - 19.05740U^2V^2 + 18.53604U^3V^2 + \\
& 18.54431U^2V^3 - 18.03391U^3V^3
\end{aligned}\right\} \quad \text{if } U>0 \ \&\ V>0 \\
& - 3.40289U^2V^2 + 3.61022U^3V^2 + 2.04219U^2V^3 \quad \rightarrow \quad \text{if } U>0 \ \&\ V\leq 0 \\
& \left.\begin{aligned}
& + 4.92677U^2V^2 + 4.41584U^3V^2 - 4.67105U^2V^3 - \\
& 3.53177U^3V^3
\end{aligned}\right\} \quad \text{if } U\leq 0 \ \&\ V>0 \\
& \left.\begin{aligned}
& - 4.32264U^2V^2 - 5.33329U^3V^2 - 12.71956U^2V^3 - \\
& 12.06456U^3V^3
\end{aligned}\right\} \quad \text{if } U\leq 0 \ \&\ V\leq 0
\end{aligned}
\tag{H-71}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.26385 + 1.13346U + 3.92425V + 0.27866U^2 + \\
& 0.71047UV - 0.17104V^2 - 0.00522U^3 - \\
& 0.97911U^2V - 0.07970V^3 + 0.13941U^3V - \\
& 0.81494UV^3 \\
& + 1.74099U^3V^2 + 1.90838U^2V^3 - 2.10909U^3V^3 \quad \rightarrow \quad \text{if } U>0 \ \&\ V>0 \\
& \left.\begin{aligned}
& + 16.99472U^2V^2 - 19.22883U^3V^2 + \\
& 18.77129U^2V^3 - 19.50126U^3V^3
\end{aligned}\right\} \quad \text{if } U>0 \ \&\ V\leq 0 \\
& \left.\begin{aligned}
& - 7.90308U^2V^2 - 11.63460U^3V^2 + 7.79940U^2V^3 + \\
& 11.88253U^3V^3
\end{aligned}\right\} \quad \text{if } U\leq 0 \ \&\ V>0 \\
& \left.\begin{aligned}
& + 13.19666U^2V^2 + 16.33949U^3V^2 + \\
& 12.71335U^2V^3 + 16.98853U^3V^3
\end{aligned}\right\} \quad \text{if } U\leq 0 \ \&\ V\leq 0
\end{aligned}
\tag{H-72}
$$

The Four-Quadrant MREs with top power 4 (ETRS89→OSGB36) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.27873 + 3.10689U - 0.38940V - 0.06132U^2 + \\
& 0.12203UV + 0.12227V^2 - 0.37704U^3 - 0.57289U^2V + \\
& 1.15597UV^2 - 0.04584V^3 + 0.14300U^4 -
\end{aligned}
\tag{H-73}
$$

$$
\begin{aligned}
&0.49873U^3V - 1.53366UV^3 - 0.32411V^4 + \\
&0.80683U^4V - 0.90120UV^4 \\
&-46.65952U^3V^2 + 32.01499U^2V^3 + 46.67181U^4V^2 + \\
&84.50908U^3V^3 - 38.12135U^2V^4 - 118.77149U^4V^3 - \\
&20.10539U^3V^4 + 61.81831U^4V^4
\end{aligned}
\Bigg\} \text{ if } U>0 \ \& \ V>0
$$

$$
\begin{aligned}
&+14.15082U^2V^2 - 77.54244U^3V^2 + 69.07137U^2V^3 + \\
&64.78916U^4V^2 - 272.28661U^3V^3 + 51.91300U^2V^4 + \\
&211.70756U^4V^3 - 198.78373U^3V^4 + 152.54299U^4V^4
\end{aligned}
\Bigg\} \text{ if } U>0 \ \& \ V\leq 0
$$

$$
\begin{aligned}
&-24.39470U^2V^2 - 92.11651U^3V^2 + 58.03307U^2V^3 - \\
&77.85274U^4V^2 + 217.12899U^3V^3 - 38.36002U^2V^4 + \\
&183.44854U^4V^3 - 135.36500U^3V^4 - 114.51258U^4V^4
\end{aligned}
\Bigg\} \text{ if } U\leq 0 \ \& \ V>0
$$

$$
\begin{aligned}
&-49.32336U^2V^2 - 126.52005U^3V^2 - 132.82103U^2V^3 - \\
&79.67970U^4V^2 - 339.29339U^3V^3 - 76.07143U^2V^4 - \\
&220.33799U^4V^3 - 205.83945U^3V^4 - 140.37624U^4V^4
\end{aligned}
\Bigg\} \text{ if } U\leq 0 \ \& \ V\leq 0
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.26229 + 1.12450U + 3.94383V + 0.25512U^2 + \\
& 0.16806UV - 0.10643V^2 - 0.01377U^3 - \\
& 3.57592U^2V + 0.26593UV^2 + 0.02330U^4 + \\
& 0.60571U^3V - 1.74417UV^3 - 0.02509V^4 + \\
& 4.04388U^4V - 0.38909UV^4 \\
& -24.24504U^2V^2 + 200.36408U^3V^2 - \\
& 114.95099U^2V^3 - 181.69039U^4V^2 - \\
& 199.85455U^3V^3 + 179.60440U^2V^4 + \\
& 327.12271U^4V^3 - 70.07436U^3V^4 - 115.16081U^4V^4
\end{aligned}
\Bigg\} \text{ if } U>0 \ \& \ V>0
$$

$$
\begin{aligned}
&+5.82887U^2V^2 - 65.83607U^3V^2 + 47.05565U^2V^3 + \\
&64.78394U^4V^2 - 258.50740U^3V^3 + 32.35122U^2V^4 + \\
&223.98160U^4V^3 - 189.64990U^3V^4 + 164.33182U^4V^4
\end{aligned}
\Bigg\} \text{ if } U>0 \ \& \ V\leq 0
$$

$$
\begin{aligned}
&-2.67738U^2V^2 - 18.05545U^3V^2 - 12.52773U^2V^3 - \\
&18.22512U^4V^2 - 13.61833U^3V^3 - 3.51837U^3V^4 - \\
&4.03493U^4V^4
\end{aligned}
\Bigg\} \text{ if } U\leq 0 \ \& \ V>0
$$

$$
\begin{aligned}
&+36.42382U^2V^2 + 99.83561U^3V^2 + \\
&123.56331U^2V^3 + 71.23058U^4V^2 + \\
&335.79256U^3V^3 + 120.34116U^2V^4 + \\
&229.33095U^4V^3 + 314.97004U^3V^4 + 208.82990U^4V^4
\end{aligned}
\Bigg\} \text{ if } U\leq 0 \ \& \ V\leq 0
\qquad \text{(H-74)}
$$

The Chebyshev MREs with top power 3 are (ETRS89→OSGB36) as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.20003 + 2.89752T_1(U) - 0.49244T_1(V) - 0.00845T_2(U) - \\
& 0.12413T_1(U)T_1(V) - 0.08617T_2(V) - 0.06800T_3(U) + \\
& 0.01016T_2(U)T_1(V) + 0.07686T_1(U)T_2(V) - 0.00868T_3(V) + \\
& 0.01488T_3(U)T_1(V) - 0.02585T_2(U)T_2(V) + \\
& 0.03816T_1(U)T_3(V) + 0.03518T_3(U)T_2(V) - \\
& 0.00424T_2(U)T_3(V) + 0.01646T_3(U)T_3(V)
\end{aligned}
\qquad \text{(H-75)}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.36794 + 1.09597T_1(U) + 3.64538T_1(V) + 0.16924T_2(U) + \\
& 0.71819T_1(U)T_1(V) - 0.04225T_2(V) - 0.06755T_3(U) - \\
& 0.05895T_2(U)T_1(V) - 0.03608T_1(U)T_2(V) + 0.04436T_3(V) + \\
& 0.05985T_3(U)T_1(V) + 0.02478T_2(U)T_2(V) - \\
& 0.04487T_1(U)T_3(V) - 0.06601T_3(U)T_2(V) + \\
& 0.09210T_2(U)T_3(V) + 0.03238T_3(U)T_3(V)
\end{aligned}
\qquad \text{(H-76)}
$$

The Chebyshev MREs with top power 4 are (ETRS89→OSGB36) as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.23752 + 2.87553T_1(U) - 0.50915T_1(V) + 0.01804T_2(U) - \\
& 0.09336T_1(U)T_1(V) - 0.04831T_2(V) - 0.07282T_3(U) - \\
& 0.01496T_2(U)T_1(V) - 0.01704T_3(V) + 0.02191T_4(U) + \\
& 0.02927T_3(U)T_1(V) - 0.01512T_2(U)T_2(V) + \\
& 0.04677T_1(U)T_3(V) + 0.00855T_4(V) - 0.04133T_4(U)T_1(V) + \\
& 0.02473T_3(U)T_2(V) - 0.01711T_2(U)T_3(V) - \\
& 0.06614T_1(U)T_4(V) + 0.04117T_4(U)T_2(V) +
\end{aligned}
\qquad \text{(H-77)}
$$

$$0.01719T_3(U)T_3(V) + 0.00526T_2(U)T_4(V) - 0.01521T_4(U)T_3(V) + 0.40280T_4(U)T_4(V)$$

$$\begin{aligned}\Delta\lambda('') = {} & 5.38246 + 1.10220T_1(U) + 3.60608T_1(V) + 0.15661T_2(U) + \\ & 0.81763T_1(U)T_1(V) - 0.06290T_2(V) - 0.02165T_3(U) - \\ & 0.10924T_2(U)T_1(V) - 0.04797T_1(U)T_2(V) + 0.00506T_3(V) - \\ & 0.04692T_4(U) + 0.06484T_3(U)T_1(V) + 0.01361T_2(U)T_2(V) + \\ & 0.02653T_1(U)T_3(V) - 0.03750T_4(V) + 0.10733T_4(U)T_1(V) + \\ & 0.03060T_3(U)T_2(V) - 0.01483T_1(U)T_4(V) - \\ & 0.02101T_4(U)T_2(V) + 0.01475T_3(U)T_3(V) - \\ & 0.05152T_4(U)T_3(V) + 0.05463T_3(U)T_4(V) + \\ & 0.02537T_4(U)T_4(V)\end{aligned} \quad \text{(H-78)}$$

The Chebyshev MREs with top power 5 are (ETRS89→OSGB36) as follows:

$$\begin{aligned}\Delta\phi('') = {} & 0.22856 + 2.88035T_1(U) - 0.47787T_1(V) + 0.00780T_2(U) - \\ & 0.10886T_1(U)T_1(V) - 0.04441T_2(V) - 0.04792T_3(U) + \\ & 0.01474T_2(U)T_1(V) - 0.02196T_1(U)T_2(V) - 0.02841T_3(V) + \\ & 0.01026T_4(U) - 0.06141T_3(U)T_1(V) + 0.01103T_2(U)T_2(V) - \\ & 0.02023T_1(U)T_3(V) + 0.02302T_5(U) + 0.08869T_4(U)T_1(V) + \\ & 0.02469T_3(U)T_2(V) + 0.02748T_2(U)T_3(V) - 0.05497T_1(U)T_4(V) - \\ & 0.01042T_5(V) - 0.14097T_5(U)T_1(V) - 0.00442T_4(U)T_2(V) - \\ & 0.01298T_3(U)T_3(V) + 0.02272T_2(U)T_4(V) - 0.01567T_1(U)T_5(V) - \\ & 0.02544T_5(U)T_2(V) + 0.09700T_4(U)T_3(V) + 0.02891T_3(U)T_4(V) - \\ & 0.09101T_5(U)T_3(V) + 0.01706T_4(U)T_4(V) + 0.00358T_3(U)T_5(V) - \\ & 0.00791T_5(U)T_4(V) + 0.04040T_4(U)T_5(V) - 0.02101T_5(U)T_5(V)\end{aligned} \quad \text{(H-79)}$$

//

$$\begin{aligned}\Delta\lambda('') = {} & 5.38116 + 1.10163T_1(U) + 3.69900T_1(V) + 0.14708T_2(U) + \\ & 0.54989T_1(U)T_1(V) - 0.06811T_2(V) - 0.02827T_3(U) + \\ & 0.10112T_2(U)T_1(V) - 0.04240T_1(U)T_2(V) + 0.07901T_3(V) - \\ & 0.03484T_4(U) - 0.03935T_3(U)T_1(V) + 0.03104T_2(U)T_2(V) - \\ & 0.12551T_1(U)T_3(V) - 0.04113T_4(V) + 0.00469T_5(U) + \\ & 0.13666T_4(U)T_1(V) + 0.02714T_3(U)T_2(V) + \\ & 0.18762T_2(U)T_3(V) - 0.01852T_1(U)T_4(V) + 0.03405T_5(V) - \\ & 0.03164T_5(U)T_1(V) - 0.00890T_4(U)T_2(V) - \\ & 0.06466T_3(U)T_3(V) + 0.03163T_2(U)T_4(V) - \\ & 0.04568T_1(U)T_5(V) - 0.00954T_5(U)T_2(V) - \\ & 0.07702T_4(U)T_3(V) + 0.05003T_3(U)T_4(V) + \\ & 0.08359T_2(U)T_5(V) - 0.07048T_5(U)T_3(V) + \\ & 0.02436T_4(U)T_4(V) - 0.03900T_3(U)T_5(V) - \\ & 0.02109T_5(U)T_4(V) - 0.02363T_4(U)T_5(V) - \\ & 0.04690T_5(U)T_5(V)\end{aligned} \quad \text{(H-80)}$$

The Chebyshev MREs with top power 6 are (ETRS89→OSGB36) as follows

$$\begin{aligned}\Delta\phi('') = {} & 0.32687 + 2.78924T_1(U) - 0.52766T_1(V) + 0.02614T_1(U)T_1(V) + \\ & 0.12307T_2(V) + 0.01565T_3(U) + 0.02497T_2(U)T_1(V) - \\ & 0.21571T_1(U)T_2(V) - 0.09443T_3(V) - 0.03025T_4(U) - \\ & 0.13625T_3(U)T_1(V) - 0.03306T_2(U)T_2(V) + 0.01577T_1(U)T_3(V) + \\ & 0.08648T_4(V) + 0.02876T_5(U) + 0.12002T_4(U)T_1(V) + \\ & 0.13431T_3(U)T_2(V) - 0.19070T_1(U)T_4(V) - 0.04540T_5(V) - \\ & 0.03154T_6(U) - 0.16545T_5(U)T_1(V) - 0.03538T_4(U)T_2(V) - \\ & 0.05412T_3(U)T_3(V) - 0.03547T_2(U)T_4(V) - 0.02048T_1(U)T_5(V) + \\ & 0.01045T_6(V) - 0.06420T_6(U)T_1(V) + 0.01238T_5(U)T_2(V) + \\ & 0.18562T_4(U)T_3(V) + 0.10685T_3(U)T_4(V) - 0.02016T_2(U)T_5(V) - \\ & 0.06623T_1(U)T_6(V) - 0.04255T_6(U)T_2(V) - 0.04603T_5(U)T_3(V) + \\ & 0.03325T_4(U)T_4(V) - 0.01715T_2(U)T_6(V) - 0.01975T_6(U)T_3(V) + \\ & 0.04231T_5(U)T_4(V) + 0.09316T_4(U)T_5(V) + 0.03104T_3(U)T_6(V) - \\ & 0.02810T_6(U)T_4(V) + 0.01388T_5(U)T_5(V) + 0.03130T_4(U)T_6(V) + \\ & 0.03002T_5(U)T_6(V) - 0.00407T_6(U)T_6(V)\end{aligned} \quad \text{(H-81)}$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.42525 + 0.92021T_1(U) + 3.62580T_1(V) + 0.25385T_2(U) + \\
& 0.77923T_1(U)T_1(V) + 0.11631T_2(V) - 011377T_3(U) - \\
& 0.06468T_2(U)T_1(V) - 0.22273T_1(U)T_2(V) + 0.00843T_3(V) + \\
& 0.12785T_4(U) + 0.08510T_3(U)T_1(V) + 0.17700T_2(U)T_2(V) - \\
& 0.04312T_1(U)T_3(V) + 0.15286T_4(V) - 0.05913T_5(U) - \\
& 0.25315T_3(U)T_2(V) + 0.01147T_5(V) + 0.11608T_6(U) + \\
& 0.05679T_5(U)T_1(V) + 0.16770T_4(U)T_2(V) - 0.01048T_3(U)T_3(V) + \\
& 0.09043T_2(U)T_4(V) + 0.01864T_1(U)T_5(V) + 0.05553T_6(V) - \\
& 0.15358T_6(U)T_1(V) - 0.15620T_5(U)T_2(V) - 0.07208T_4(U)T_3(V) - \\
& 0.21925T_3(U)T_4(V) + 0.00496T_2(U)T_5(V) + 0.00399T_1(U)T_6(V) + \\
& 0.19448T_6(U)T_2(V) + 0.11382T_5(U)T_3(V) + 0.01533T_4(U)T_4(V) - \\
& 0.01191T_3(U)T_5(V) + 0.02787T_2(U)T_6(V) - 0.16211T_5(U)T_4(V) - \\
& 0.03089T_4(U)T_5(V) - 0.07863T_3(U)T_6(V) + 0.09984T_6(U)T_4(V) + \\
& 0.03212T_5(U)T_5(V) - 0.01181T_4(U)T_6(V) + 0.00350T_6(U)T_5(V) - \\
& 0.05723T_5(U)T_6(V) + 0.03056T_6(U)T_6(V)
\end{aligned}
\tag{H-82}
$$

The trimmed Ordinary MREs with top power 6 (ETRS89→OSGB36) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.28493 + 3.15216U - 0.36650V - 0.26190U^2 - 0.74750U^3 - \\
& 0.37527U^2V + 1.20355UV^2 - 0.76685V^3 + 0.68510U^4 - \\
& 0.68979U^3V - 0.67301U^2V^2 - 0.58619UV^3 - 1.26014V^4 + \\
& 0.46438U^5 + 0.21718U^4V - 4.53429U^3V^2 + 7.12046U^2V^3 - \\
& 2.12697UV^4 + 0.89376V^5 - 0.42117U^6 + 0.63487U^5V - \\
& 1.40554U^4V^2 + 4.70583U^3V^3 + 8.66415U^2V^4 + \\
& 1.57227V^6 + 3.35246U^5V^2 - 4.81115U^4V^3 + \\
& 17.85816U^3V^4 - 10.67462U^2V^5 + 2.06358U^6V^2 - \\
& 4.34005U^5V^3 - 8.20442U^4V^4 - 9.16093U^2V^6 - \\
& 1.67983U^6V^3 - 15.41019U^5V^4 + 9.99949U^4V^5 - \\
& 14.21089U^3V^6 + 13.41034U^4V^6 + 13.50978U^5V^6 - \\
& 4.73632U^6V^6
\end{aligned}
\tag{H-83}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.26196 + 1.12123U + 3.89689V + 0.27388U^2 + \\
& 0.14114UV - 0.23622V^2 - 3.03922U^2V + \\
& 0.36469V^3 + 2.98914U^3V + 1.38780U^2V^2 - \\
& 0.31057UV^3 + 0.87690V^4 + 7.05238U^4V - \\
& 1.80684U^2V^3 - 0.50145V^5 - 2.21122U^5V - \\
& 1.06563U^4V^2 - 9.19902U^2V^4 + 1.18117UV^5 - \\
& 1.03582V^6 - 4.37667U^6V + 1.51553U^4V^3 + \\
& 2.60320U^2V^5 + 20.54743U^4V^4 - 4.58824U^3V^5 - \\
& 10.78951U^2V^6 - 2.26375U^4V^5 - 13.15452U^6V^4 + \\
& 3.66625U^5V^5 - 26.07461U^4V^6 - 0.36381U^5V^6 + \\
& 17.51258U^6V^6
\end{aligned}
\tag{H-84}
$$

The trimmed North/South MREs with top power 5 (ETRS89→OSGB36) are as follows:

$$
\begin{aligned}
\Delta\phi('') = {} & 0.28274 - 0.38567V + 0.13378V^2 - 0.36550V^4 - \\
& 0.06596V^5 + 3.22429U - 1.44981UV^3 + 0.42571UV^4 \\
& - 0.65998U^2 - 2.42930U^2V^2 + 3.98034U^2V^3 - \\
& 3.87840U^3V + 9.04418U^3V^2 - 2.52425U^3V^3 + \\
& 0.31318U^4 + 7.87398U^4V - 13.01017U^4V^2 - \\
& 4.24886U^5V + 6.12799U^5V^2 \qquad \text{if } U>0 \\
& + 0.63970U^2 - 24.86874U^2V^2 - 10.12025U^2V^3 + \\
& 44.34803U^2V^4 + 2.34166U^3 - 88.46987U^3V^2 - \\
& 32.25022U^3V^3 + 150.72671U^3V^4 + 20.59973U^3V^5 + \\
& 5.00843U^4 + 0.73673U^4V - 108.58055U^4V^2 - \\
& 41.89887U^4V^3 + 176.54452U^4V^4 + 44.03018U^4V^5 + \\
& 2.94738U^5 + 0.77772U^5V - 44.63144U^5V^2 - \\
& 18.45018U^5V^3 + 69.22826U^5V^4 + 23.39180U^5V^5 \qquad \text{if } U\leq 0
\end{aligned}
\tag{H-85}
$$

$$
\begin{aligned}
\Delta\lambda('') = {} & 5.26546 + 3.91765V - 0.19198V^2 + 1.10162U + \\
& 0.32742UV + 0.84945UV^2 - 0.76076UV^3 - \\
& 0.8207UV^4 \\
& - 3.09588U^2V + 2.91768U^2V^3 + 0.95188U^2V^4 + \\
& 3.35184U^3V + 3.11526U^4 - 5.09247U^4V^2 - \\
& 2.25876U^4V^3 - 2.83209U^5 + 4.17931U^5V^2 \\
& + 9.59989U^2V^2 - 12.38098U^2V^3 - 3.79466U^2V^4 - \\
& 0.59667U^3 + 15.63958U^3V + 26.3831U^3V^2 - \\
& 42.87075U^3V^3 - 3.29270U^3V^4 - 0.35837U^4 + \\
& 28.33820U^4V + 31.84657U^4V^2 - 51.55637U^4V^3 + \\
& 13.70320U^5V + 14.12399U^5V^2 - 20.60202U^5V^3
\end{aligned}
\qquad
\begin{cases}
\text{if } U>0 \\
\text{if } U\leq 0
\end{cases}
\qquad \text{(H-86)}
$$

APPENDIX I: CONTOUR MAPS FROM CASE STUDIES

This appendix contains contour maps of some of the datum shifts derived during the case studies recorded in Chapters 8, 12 and 13. They were generated by LSS. MREs mentioned were all obtained by eliminating ratios lower than one (ERLTO), except where the word "trimmed" indicates that further elimination of terms occurred.

I.1 Western Australia (AGD84 to GDA94)

The contour maps for Western Australia are split between those derived from multiple regression equations and those derived from composite methods.

I.1.1 MREs in Western Australia

The contour maps arising from MREs described in Section 8.2 are listed below.

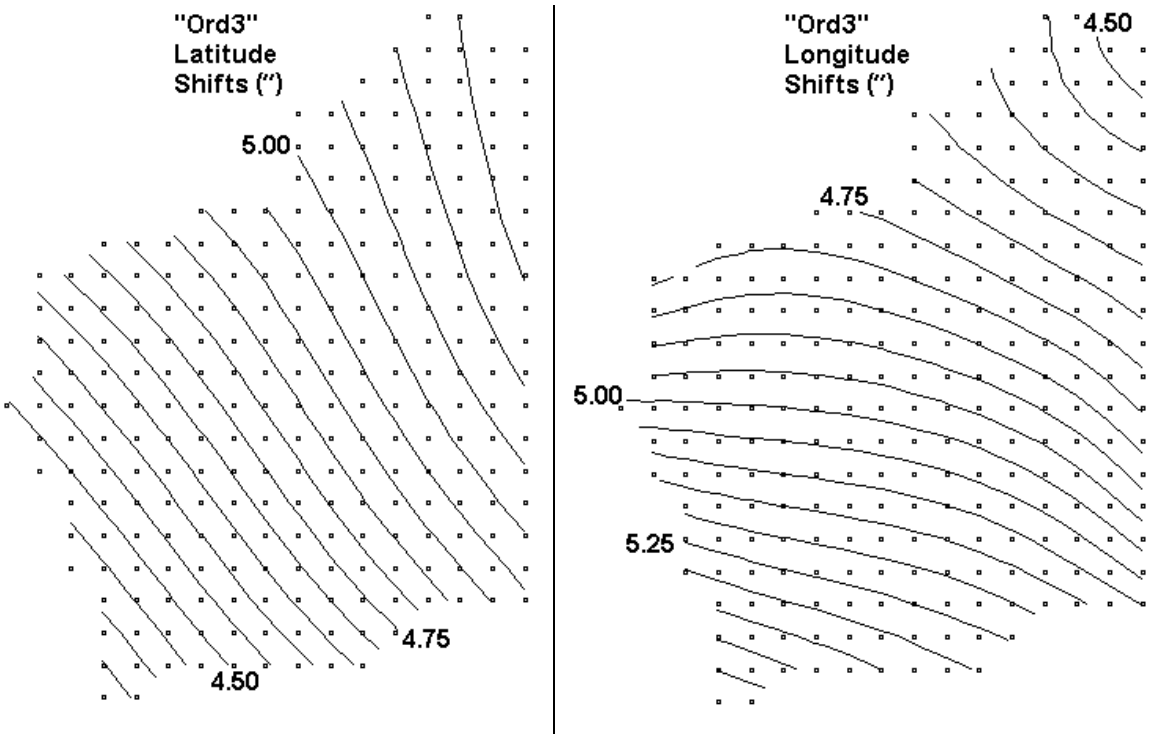


Figure I-1: Horizontal datum shifts for AGD84→GDA94 arising from the Ordinary MREs with top power 3.

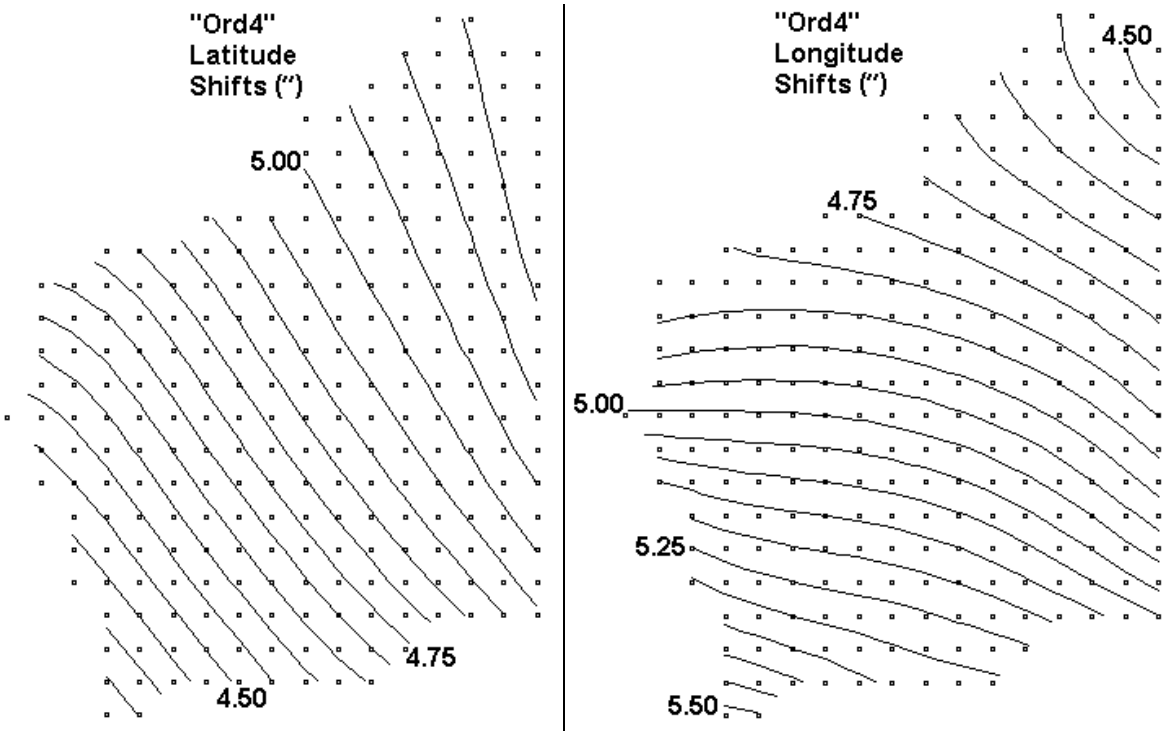


Figure I-2: Horizontal datum shifts for AGD84→GDA94 arising from the Ordinary MREs with top power 4.

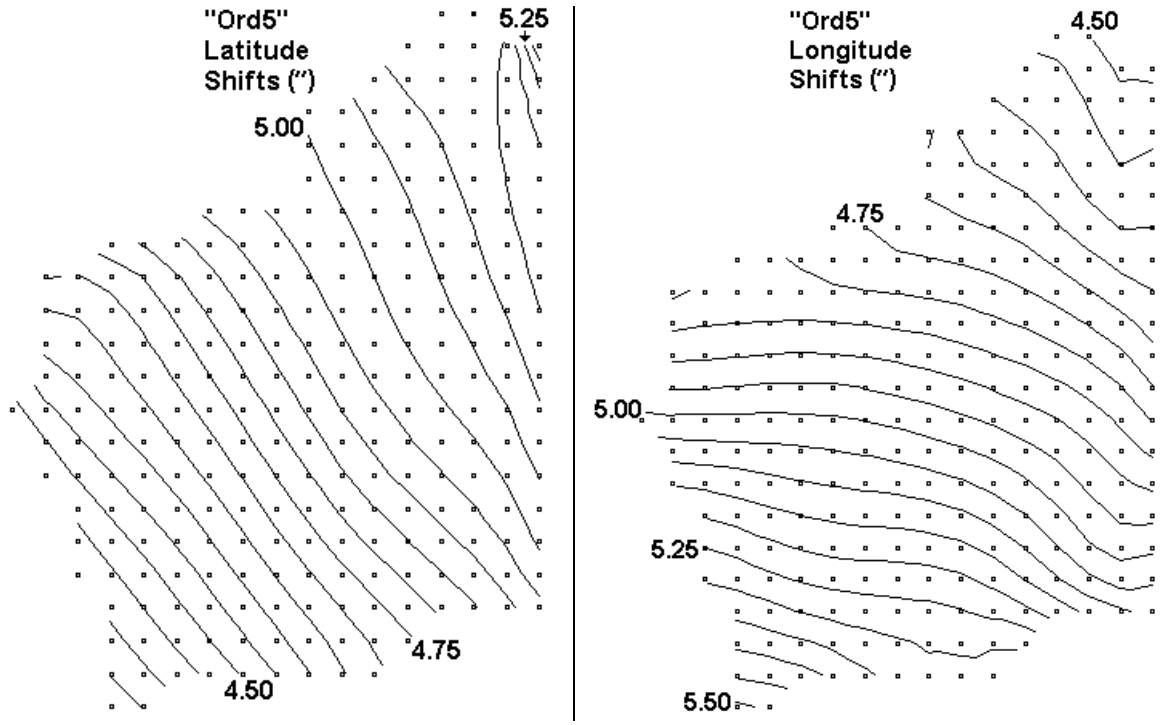


Figure I-3: Horizontal datum shifts for AGD84→GDA94 arising from the Ordinary MREs with top power 5.

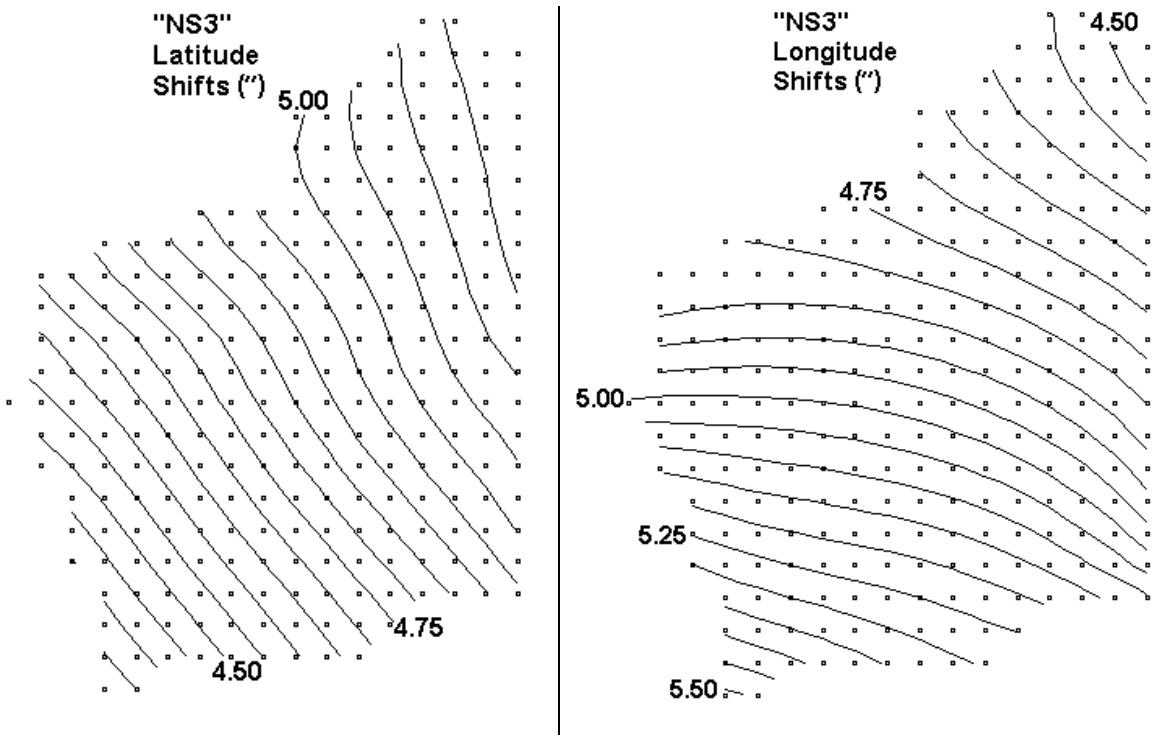


Figure I-4: Horizontal datum shifts for AGD84→GDA94 arising from the North/South MREs with top power 3.

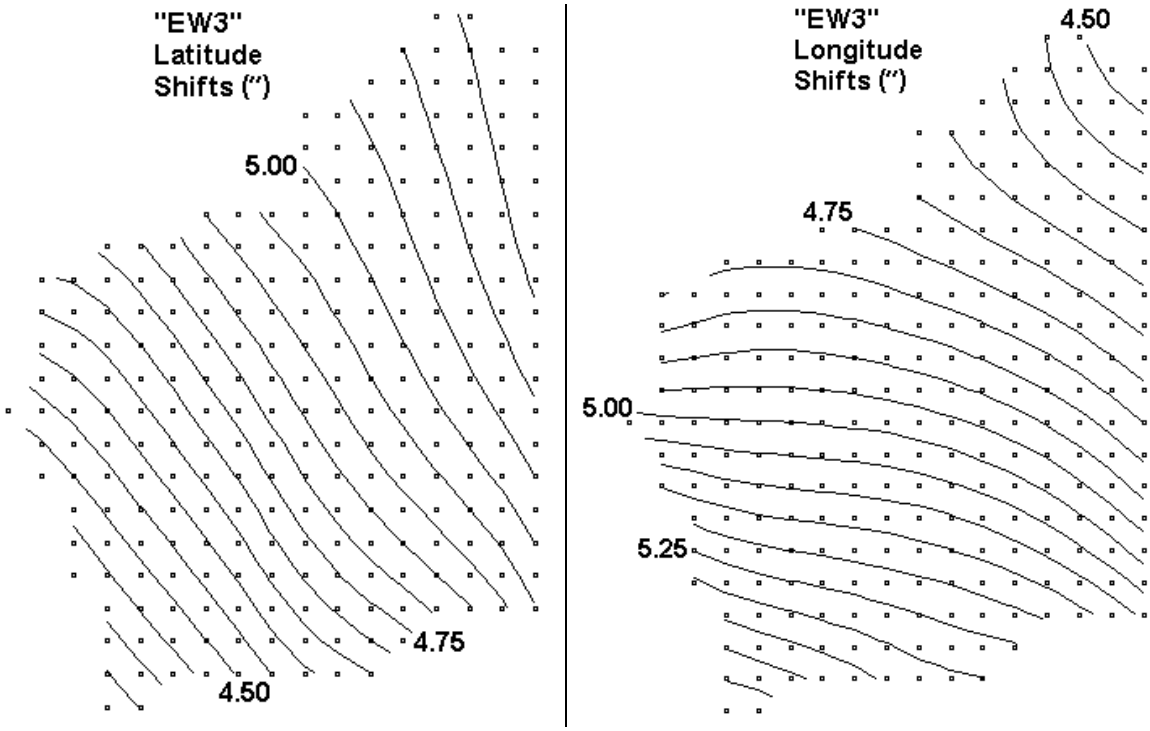


Figure I-5: Horizontal datum shifts for AGD84→GDA94 arising from the East/West MREs with top power 3.

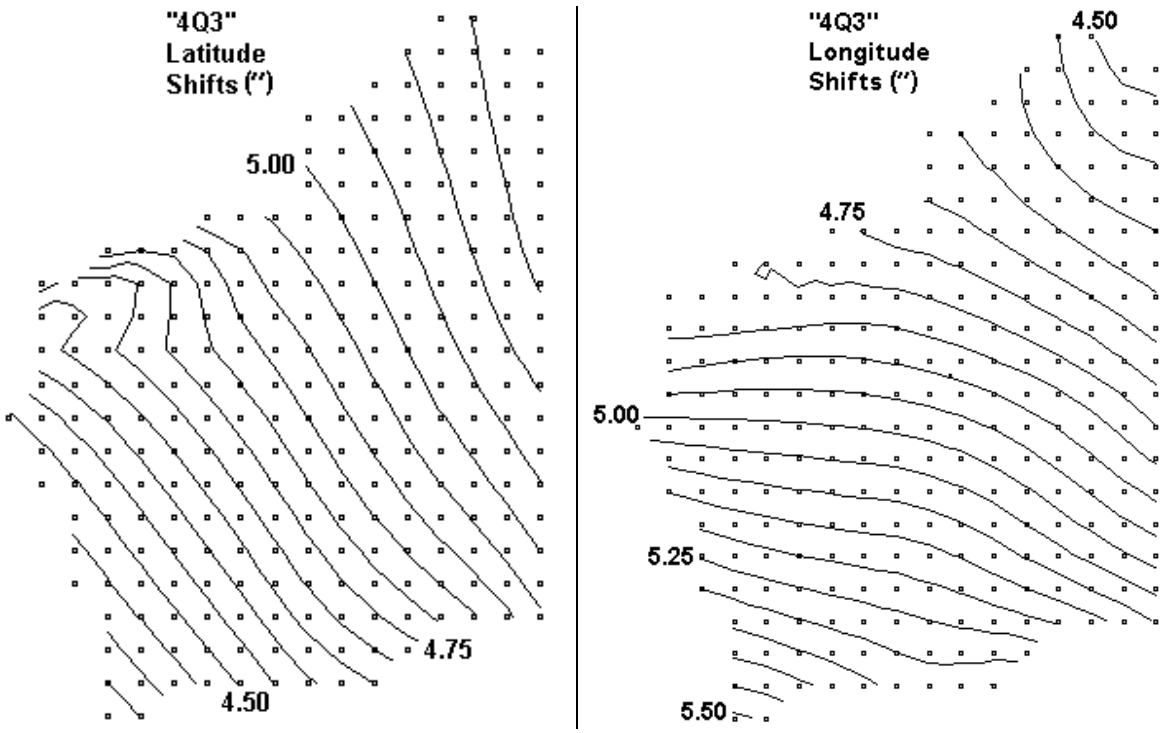


Figure I-6: Horizontal datum shifts for AGD84→GDA94 arising from the Four-Quadrant MREs with top power 3.

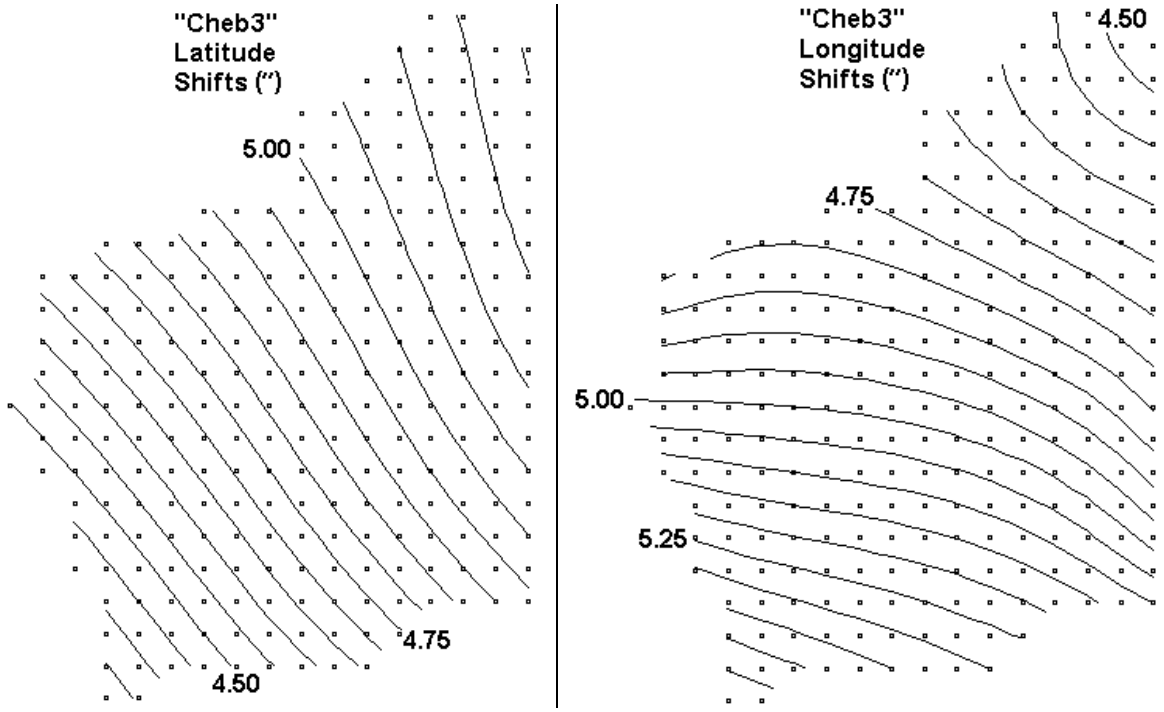


Figure I-7: Horizontal datum shifts for AGD84→GDA94 arising from the Chebyshev MREs with top power 3.

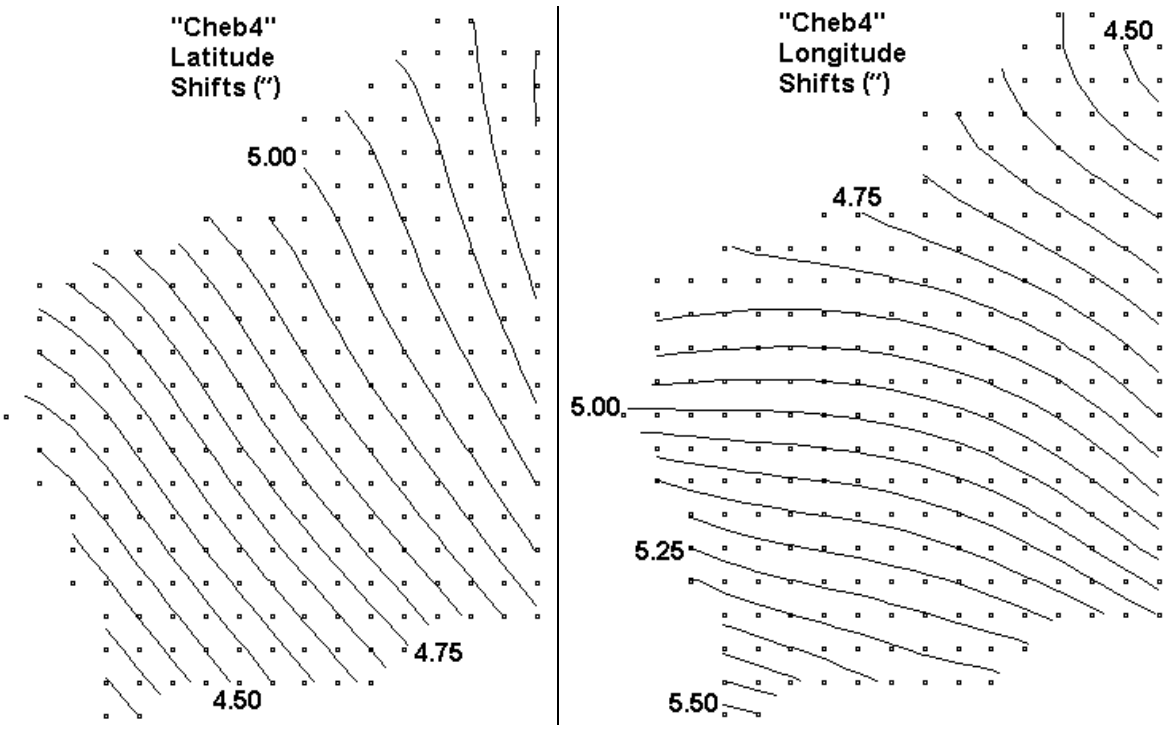


Figure I-8: Horizontal datum shifts for AGD84→GDA94 arising from the Chebyshev MREs with top power 4.

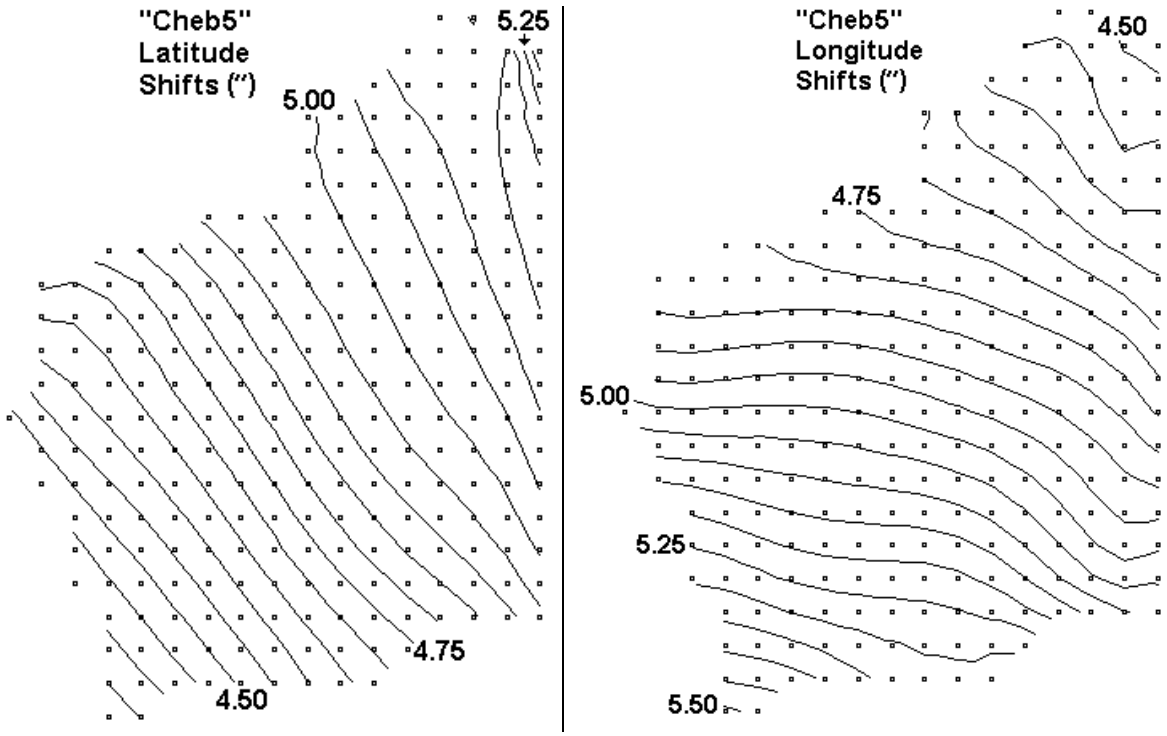


Figure I-9: Horizontal datum shifts for AGD84→GDA94 arising from the Chebyshev MREs with top power 5.

I.1.2 Composite Methods in Western Australia

The contour maps arising from composite methods described in Section 8.3 are listed below.

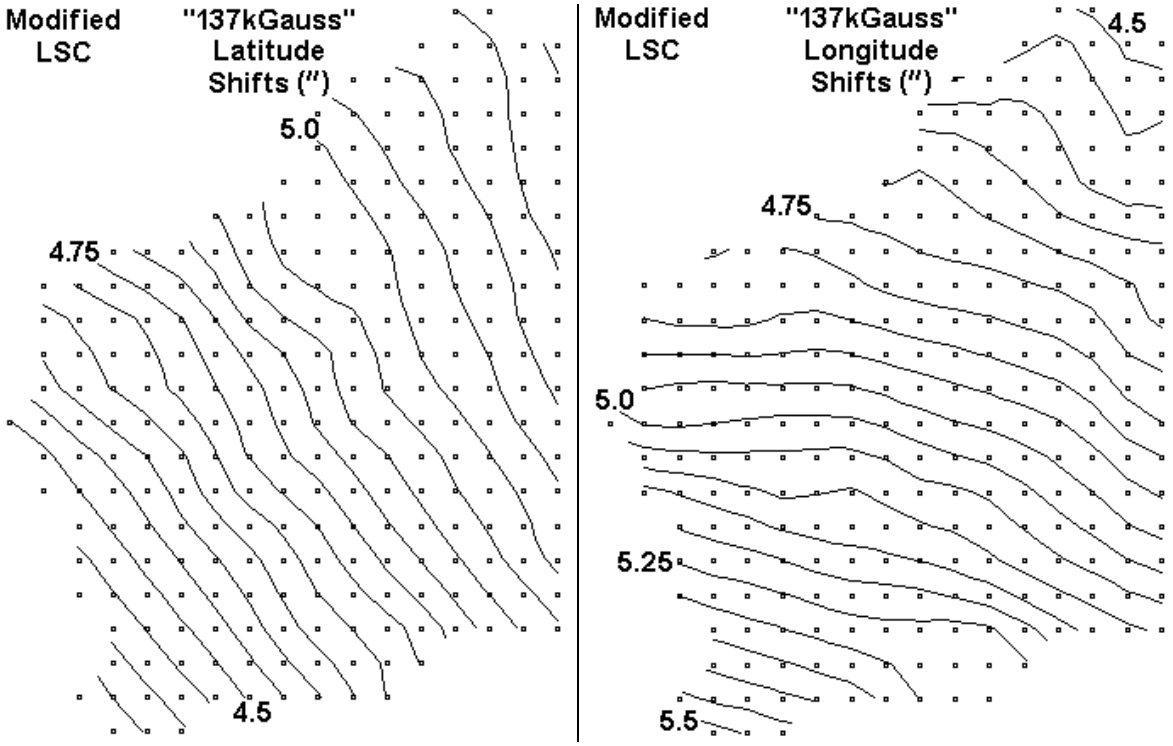


Figure I-10: Horizontal datum shifts for AGD84→GDA94 arising from Modified Least-Squares Collocation with the Gaussian covariance function and Correlation Length=137028m.

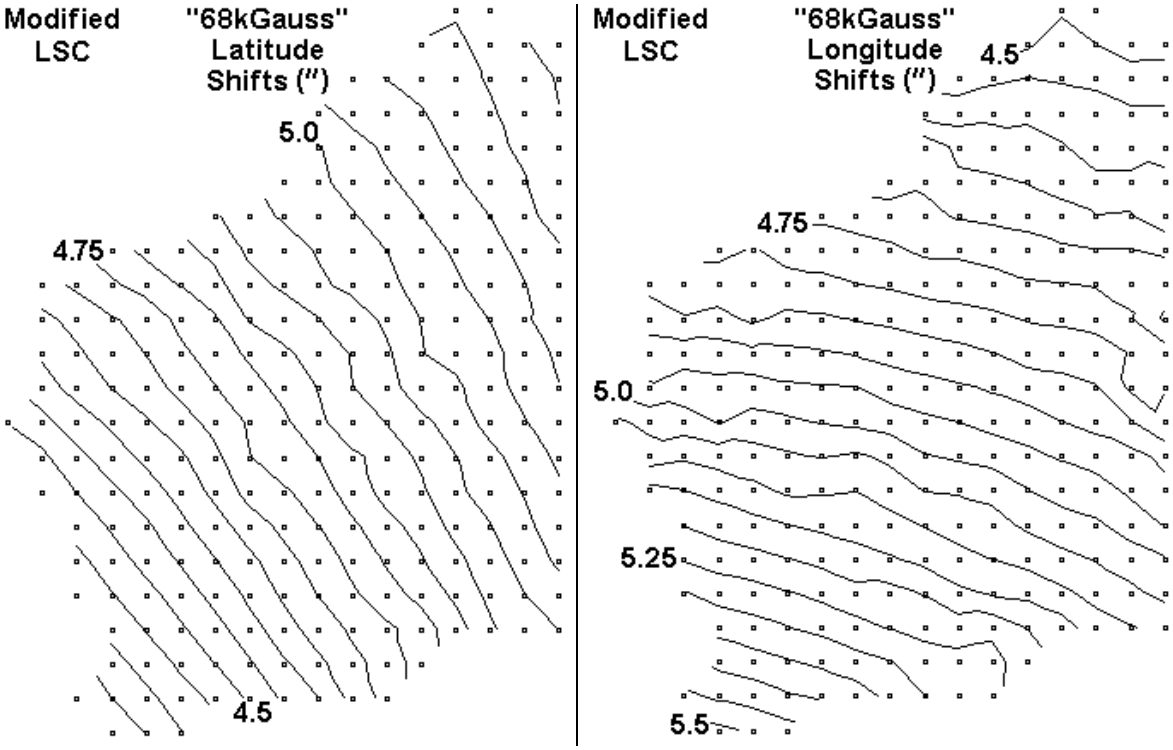


Figure I-11: Horizontal datum shifts for AGD84→GDA94 arising from Modified Least-Squares Collocation with the Gaussian covariance function and Correlation Length=68514m.

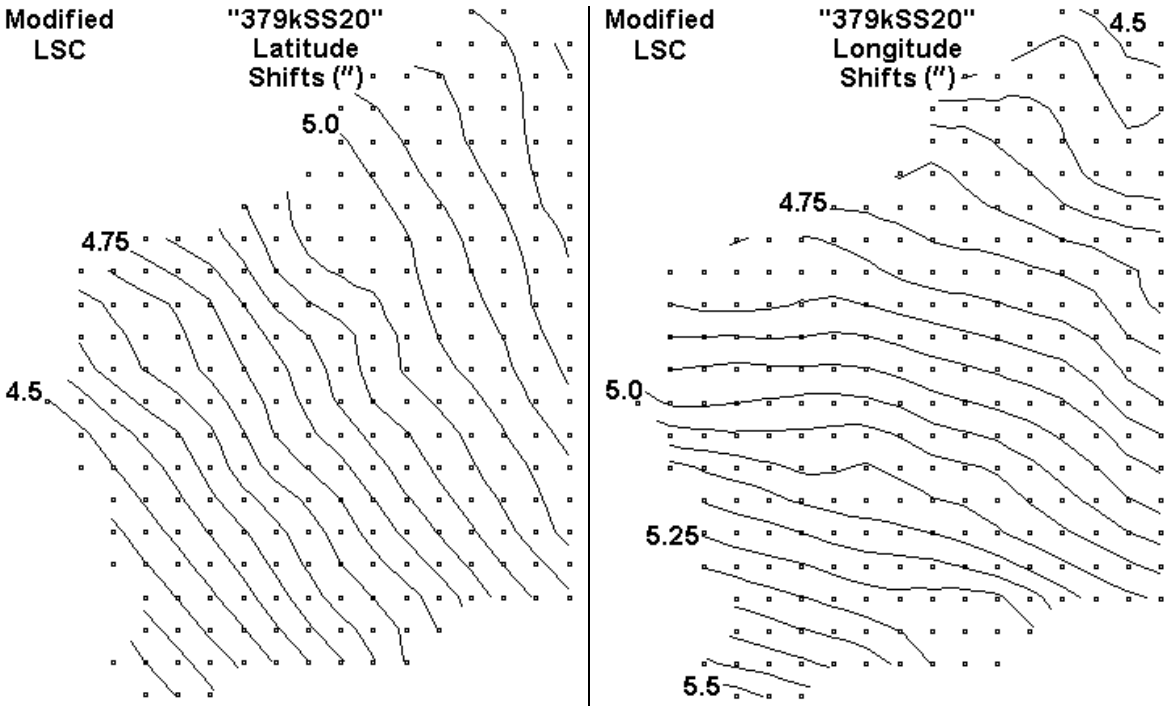


Figure I-12: Horizontal datum shifts for AGD84→GDA94 arising from Modified Least-Squares Collocation with the SS20 covariance function and r_{max}=379395m.

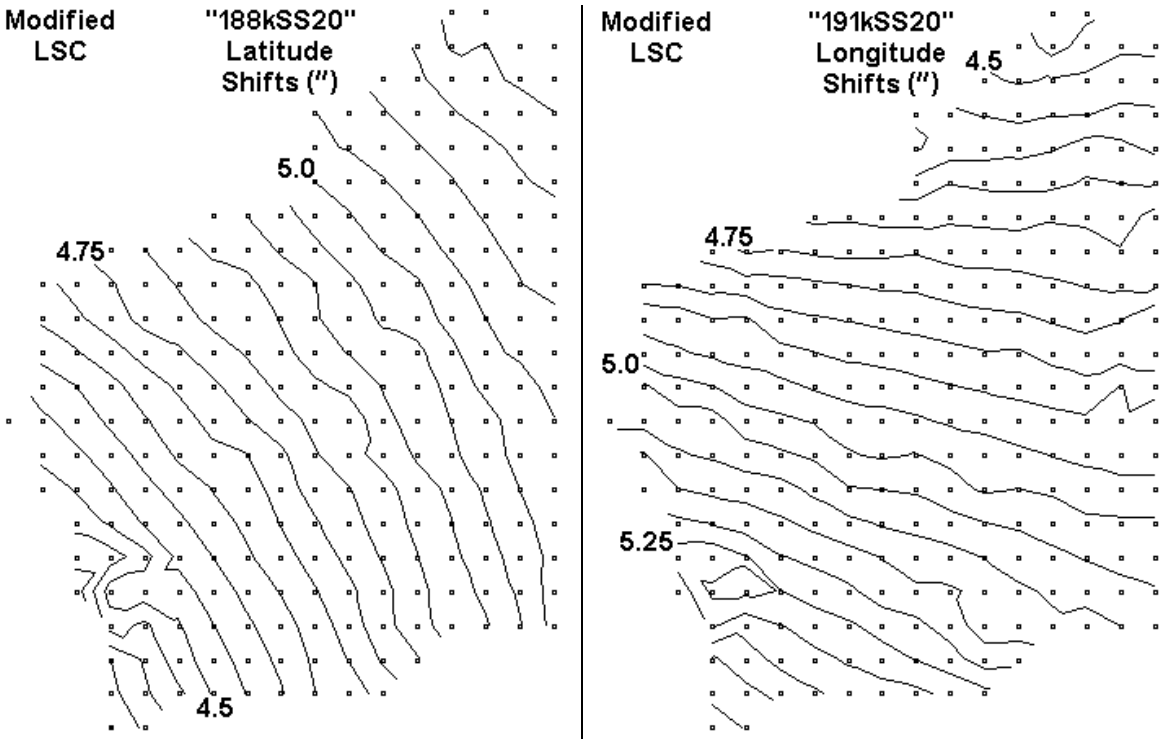


Figure I-13: Horizontal datum shifts for AGD84→GDA94 arising from Modified Least-Squares Collocation with the SS20 covariance function and r_{max}=188358m for latitude and r_{max}=191513m for longitude.

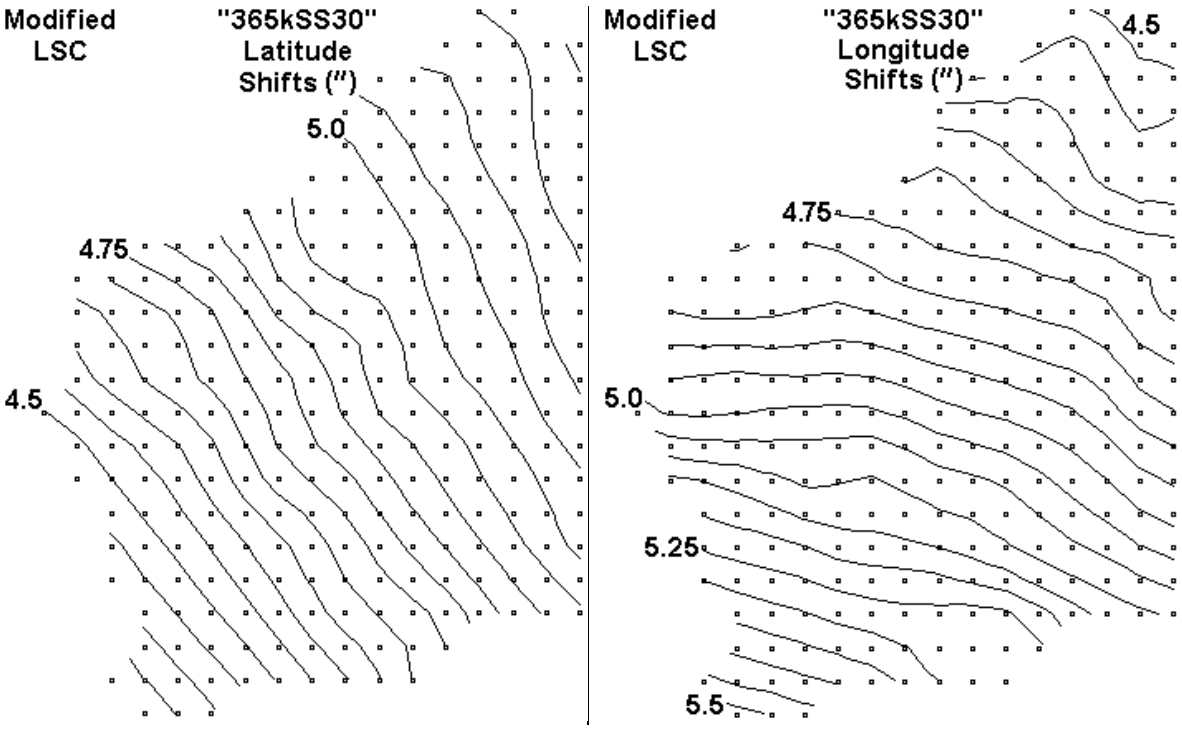


Figure I-14: Horizontal datum shifts for AGD84→GDA94 arising from Modified Least-Squares Collocation with the SS30 covariance function and r_{max}=365690m.

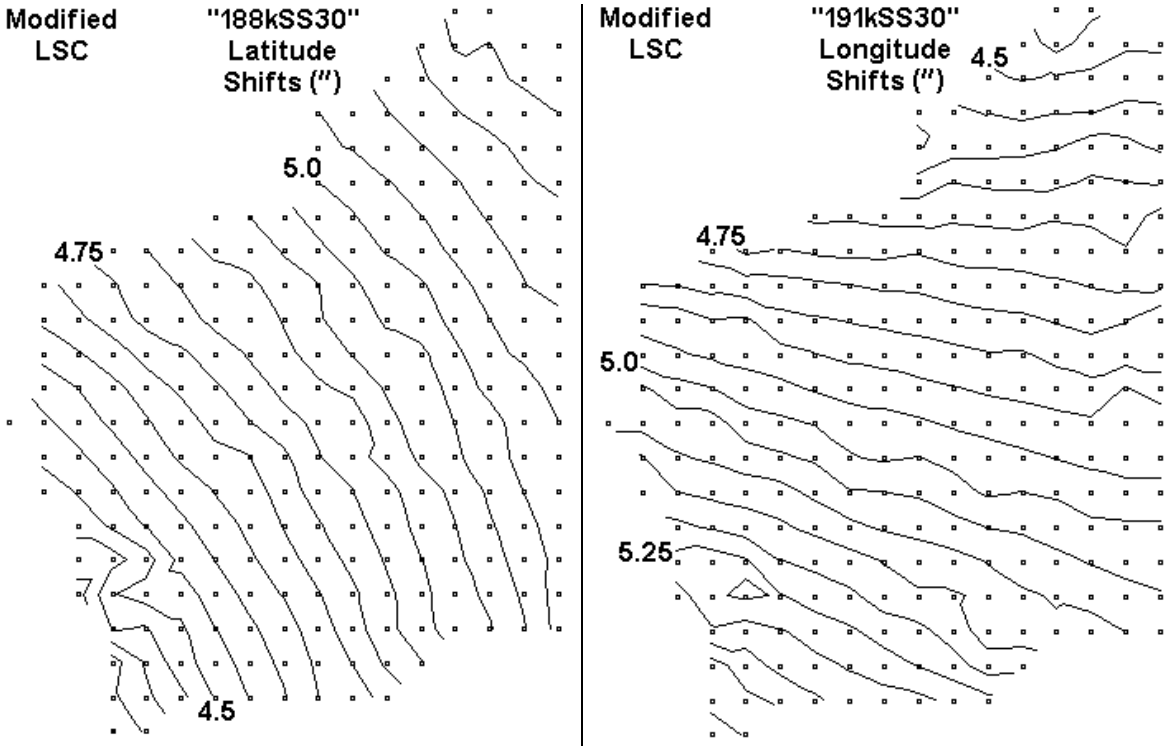


Figure I-15: Horizontal datum shifts for AGD84→GDA94 arising from Modified Least-Squares Collocation with the SS20 covariance function and r_{max}=188358m for latitude and r_{max}=191513m for longitude.

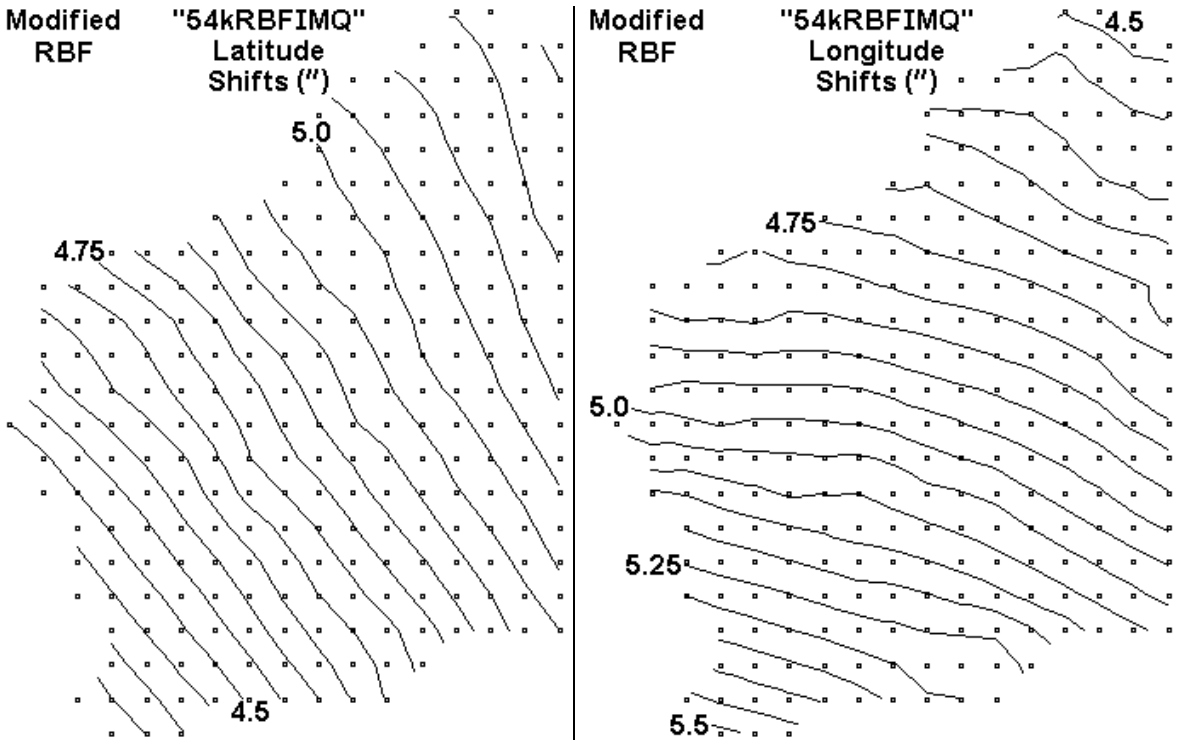


Figure I-16: Horizontal datum shifts for AGD84→GDA94 arising from Modified Radial Basis Function "Inverse Multiquadric" with shaping parameter 54347m.

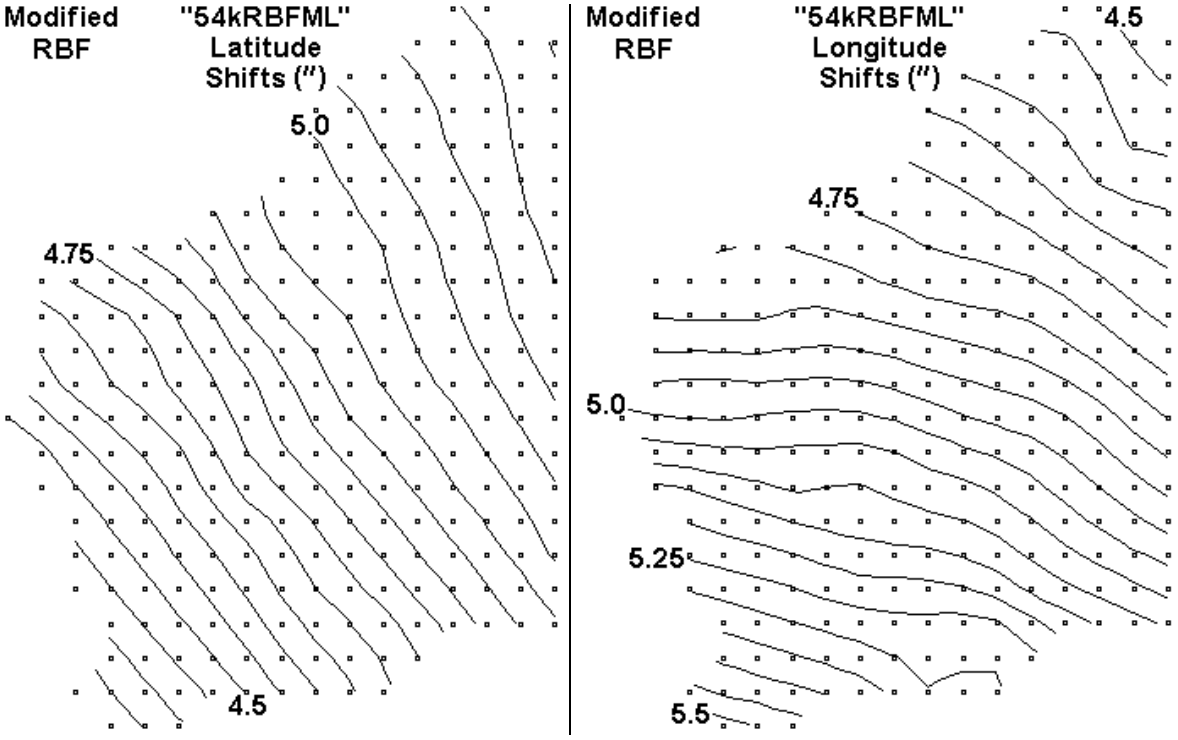


Figure I-17: Horizontal datum shifts for AGD84→GDA94 arising from Modified Radial Basis Function "Multilog" with shaping parameter 54347m.

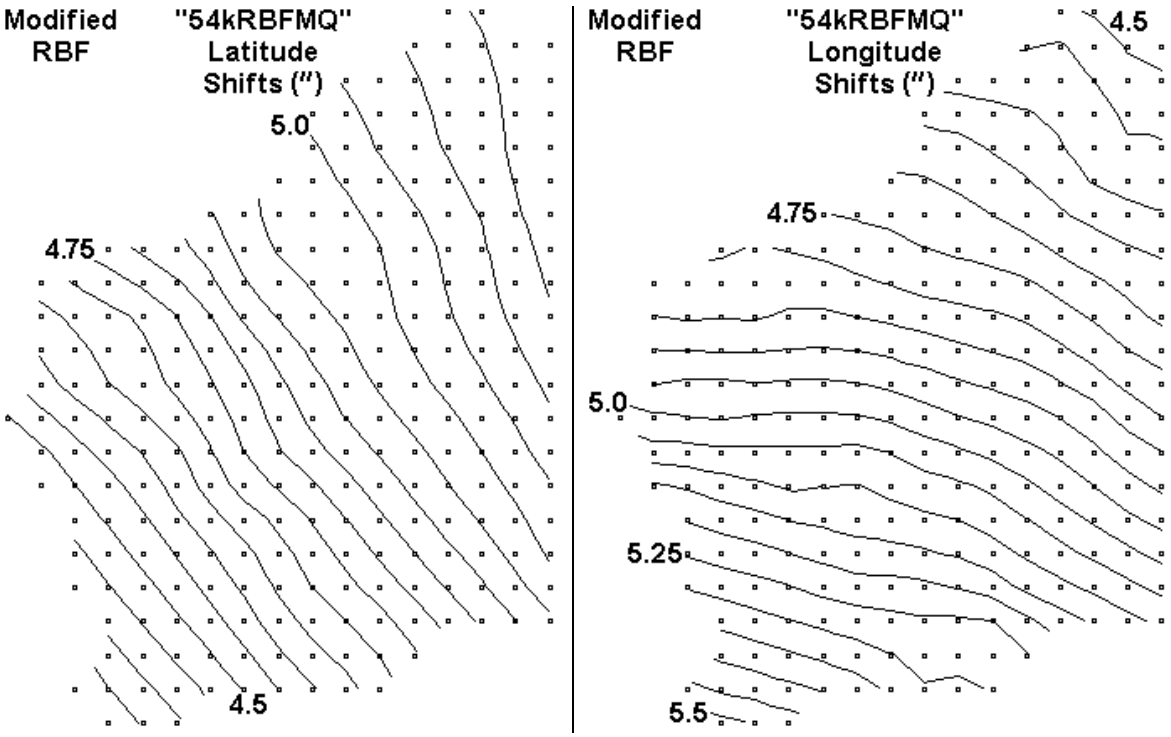


Figure I-18: Horizontal datum shifts for AGD84→GDA94 arising from Modified Radial Basis Function "Multiquadric" with shaping parameter 54347m.

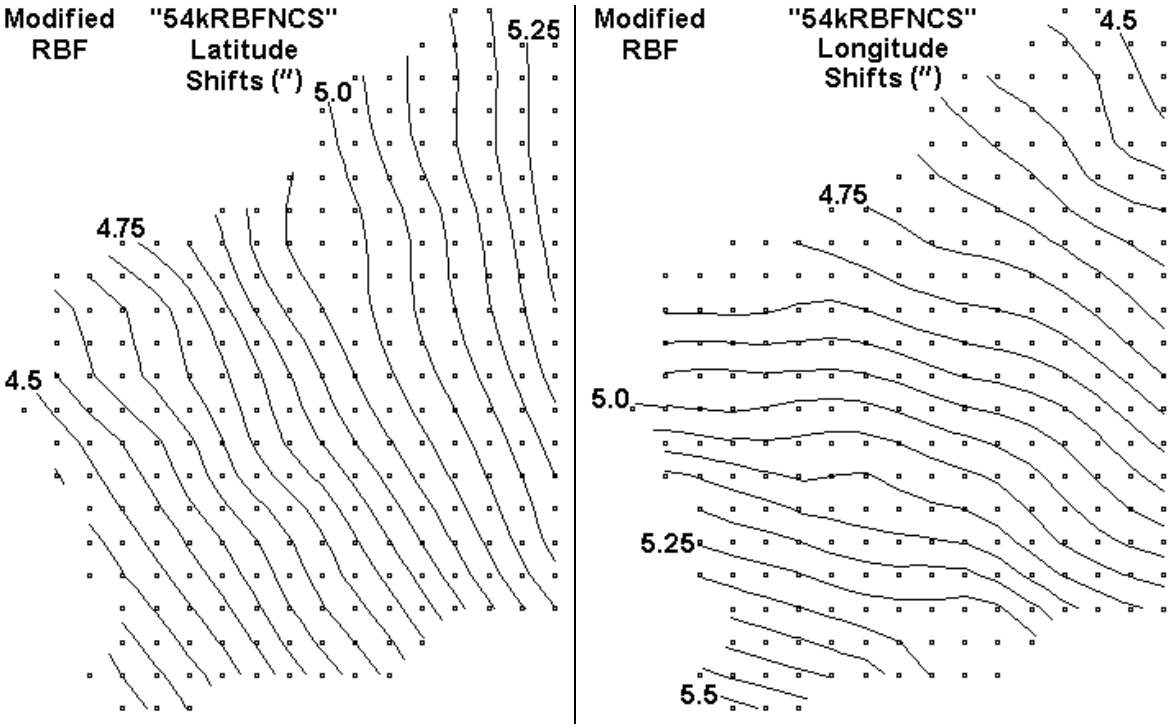


Figure I-19: Horizontal datum shifts for AGD84→GDA94 arising from Modified Radial Basis Function "Natural Cubic Spline" with shaping parameter 54347m.

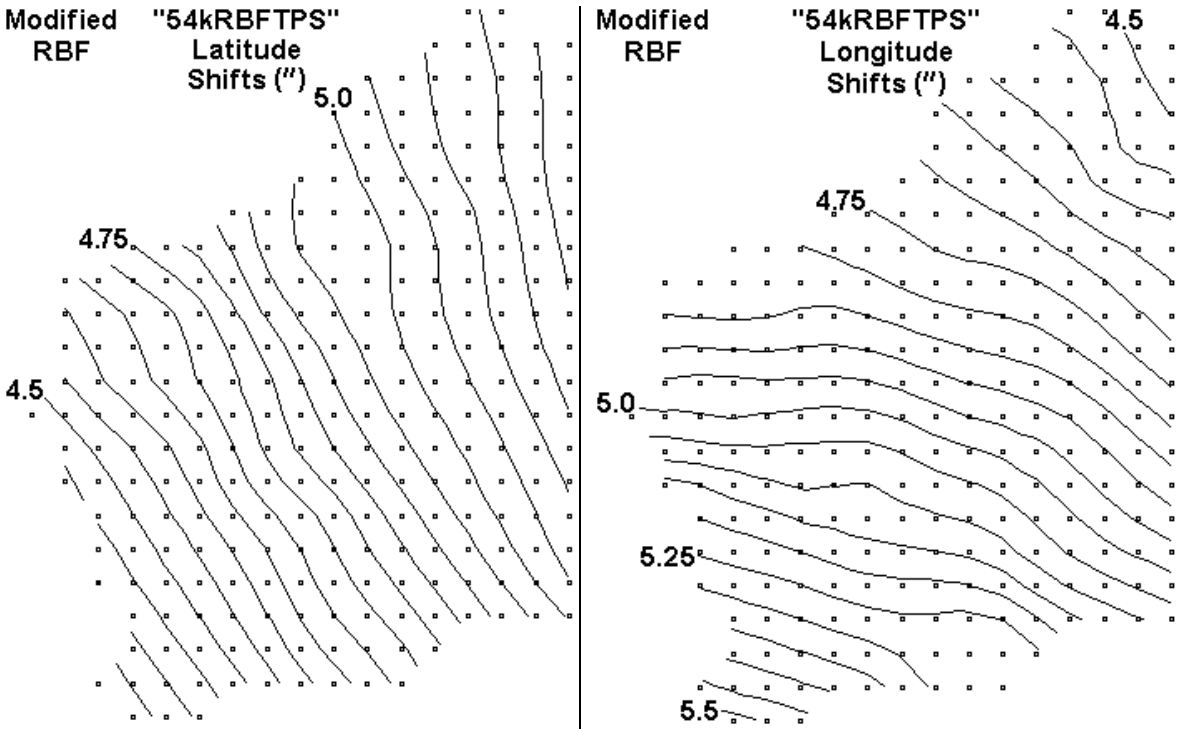


Figure I-20: Horizontal datum shifts for AGD84→GDA94 arising from Modified Radial Basis Function "Thin Plate Spline" with shaping parameter 54347m.

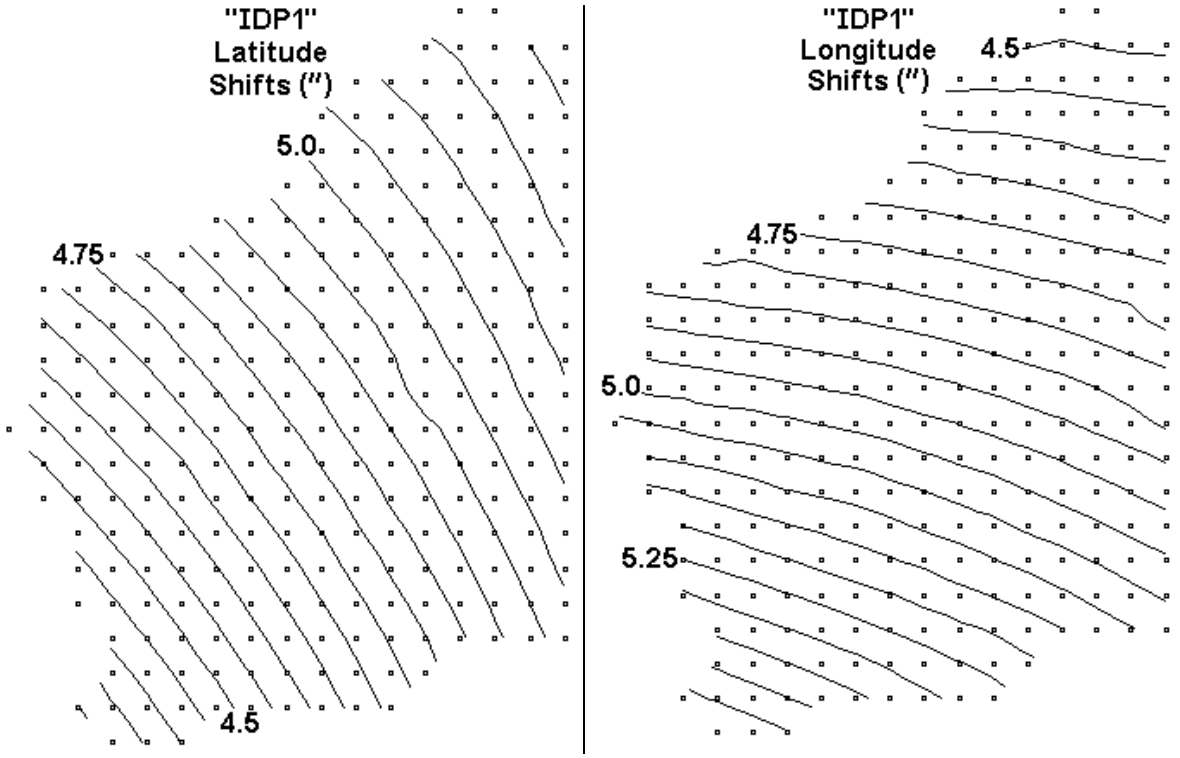


Figure I-21: Horizontal datum shifts for AGD84→GDA94 arising from Inverse Distance To The Power 1.

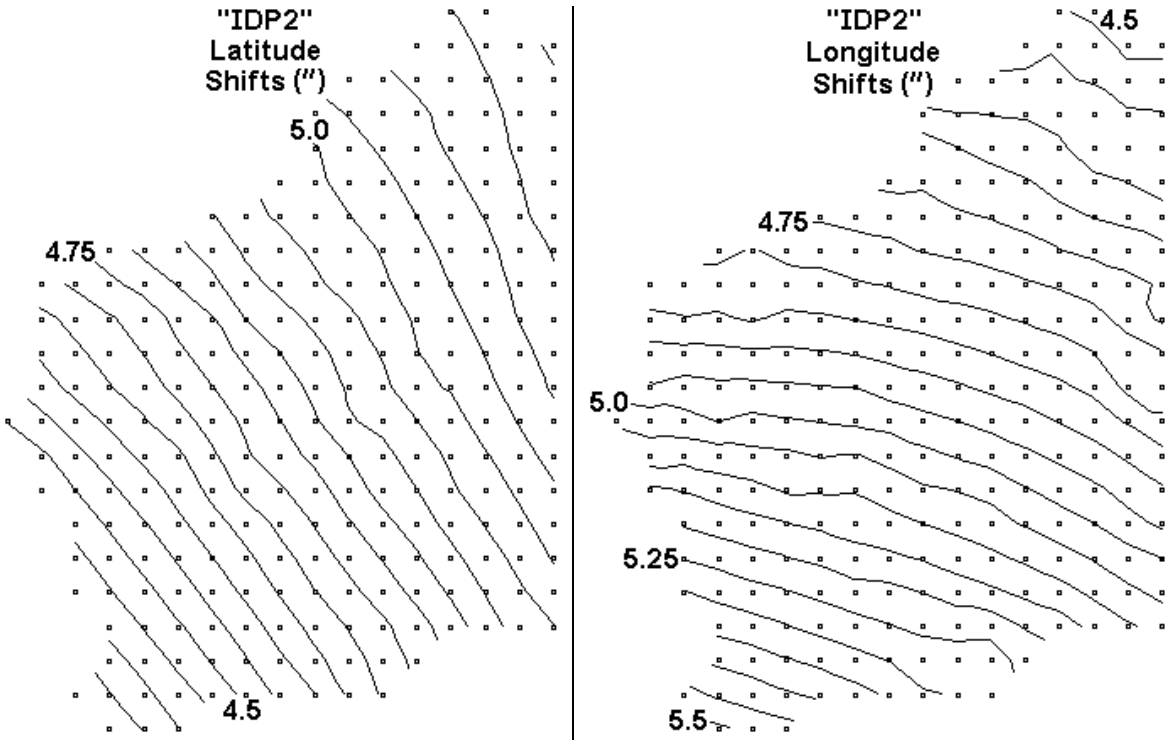


Figure I-22: Horizontal datum shifts for AGD84→GDA94 arising from Inverse Distance To The Power 2.

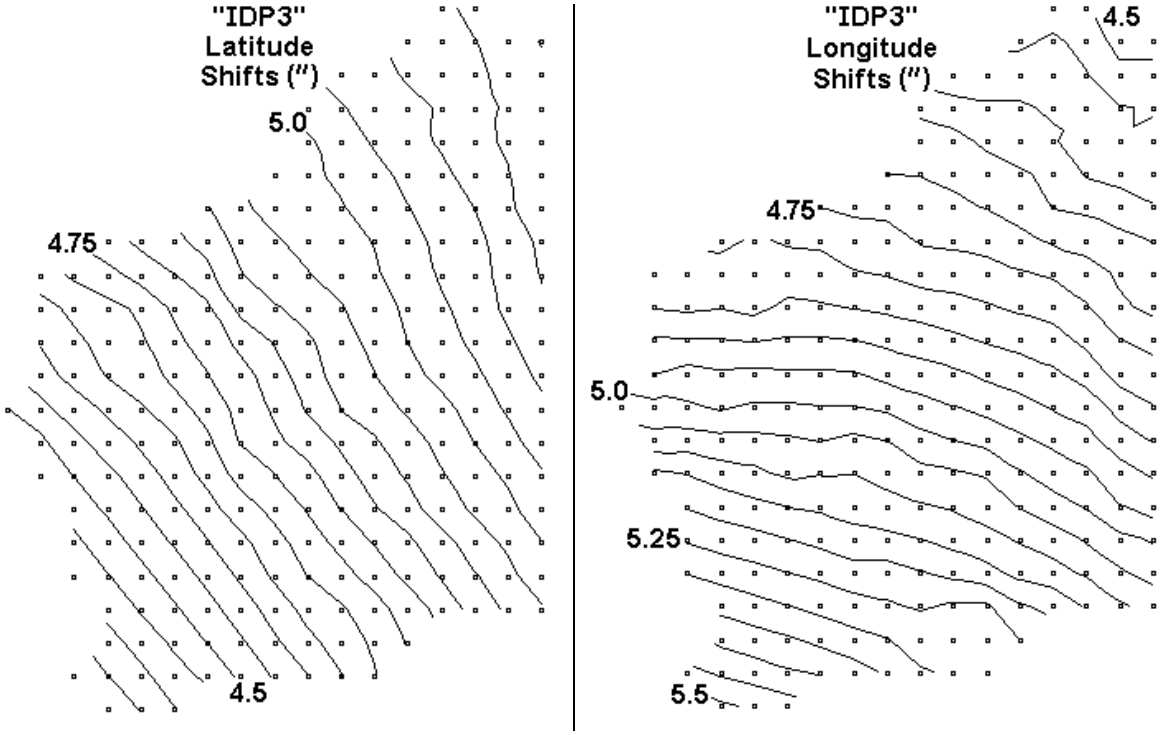


Figure I-23: Horizontal datum shifts for AGD84→GDA94 arising from Inverse Distance To The Power 3.

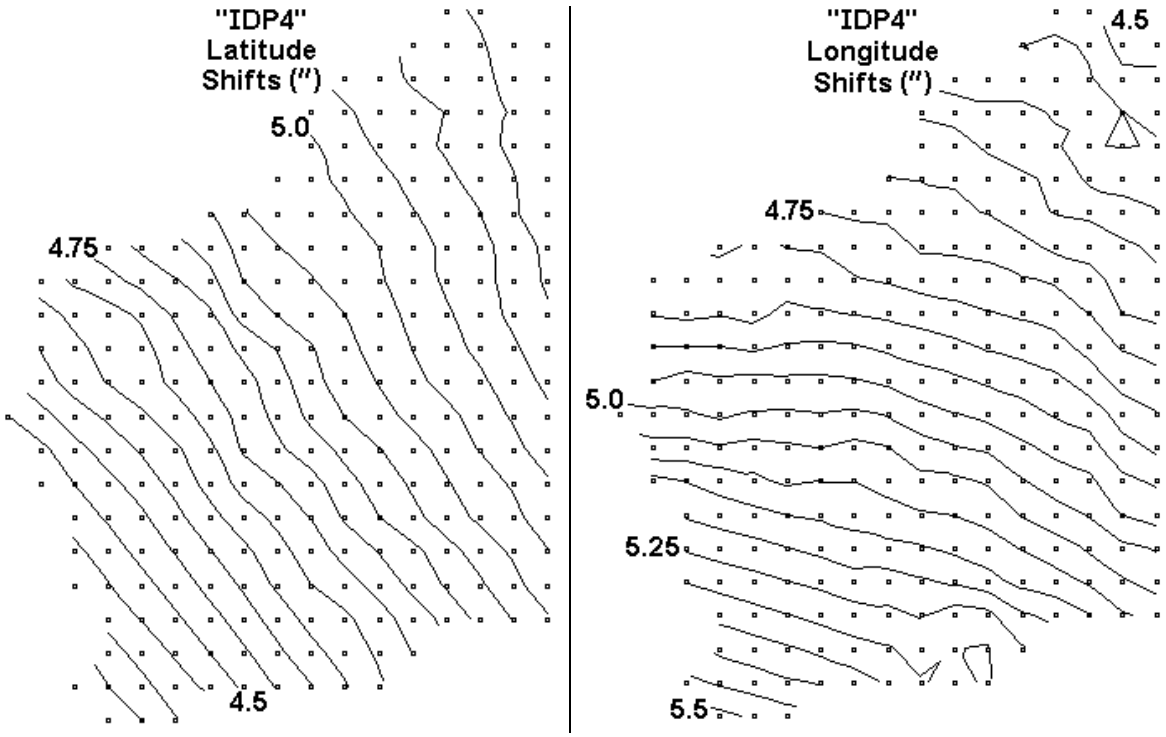


Figure I-24: Horizontal datum shifts for AGD84→GDA94 arising from Inverse Distance To The Power 4.

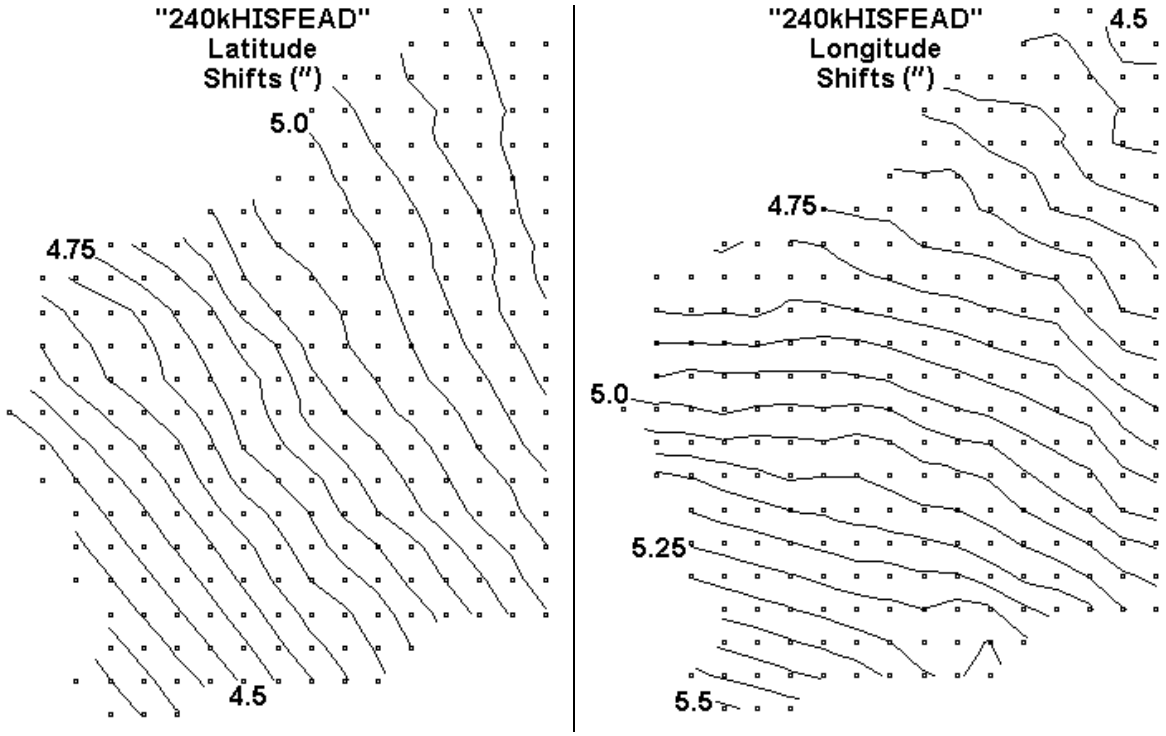


Figure I-25: Horizontal datum shifts for AGD84→GDA94 arising from HISFEAD with r_{max}=240000m.

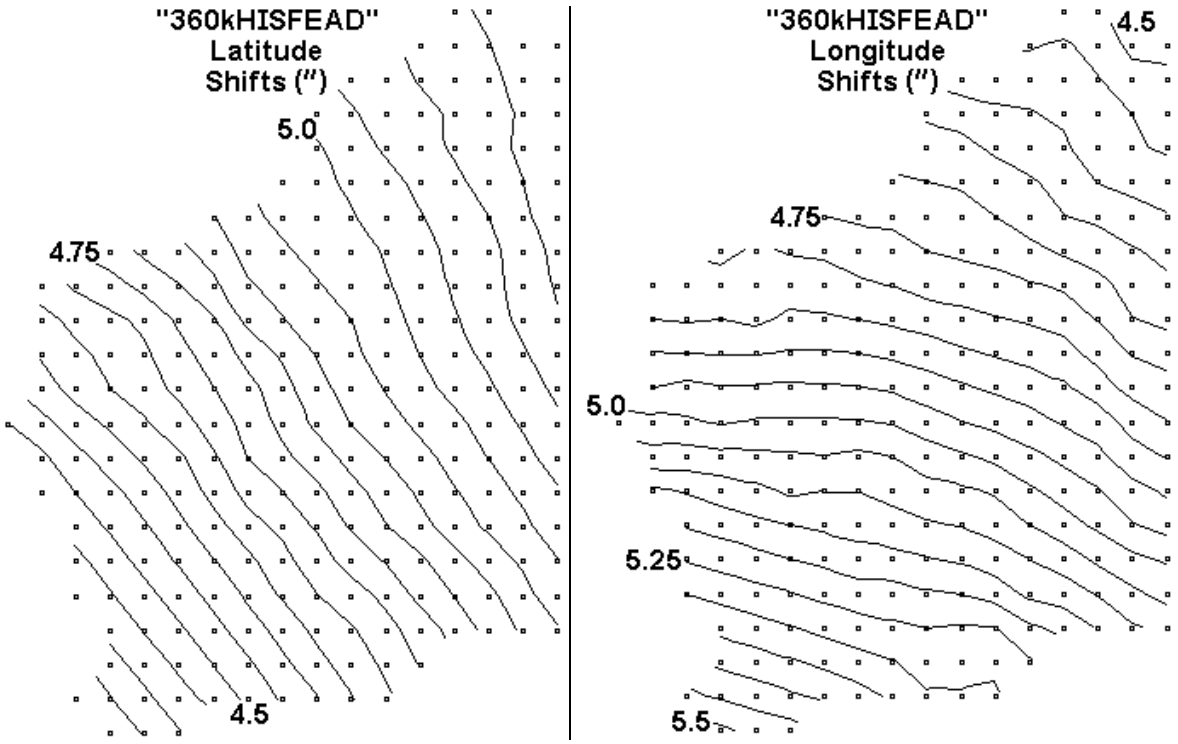


Figure I-26: Horizontal datum shifts for AGD84→GDA94 arising from HISFEAD with r_{max}=360000m.

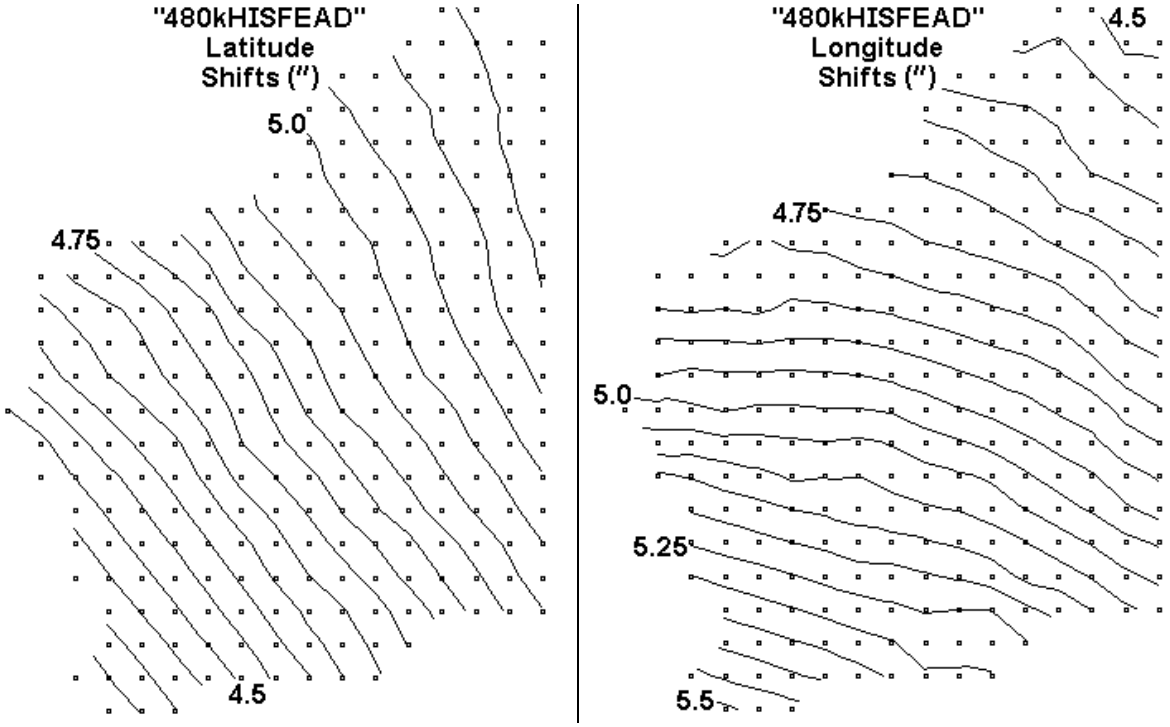


Figure I-27: Horizontal datum shifts for AGD84→GDA94 arising from HISFEAD with r_{max}=480000m.

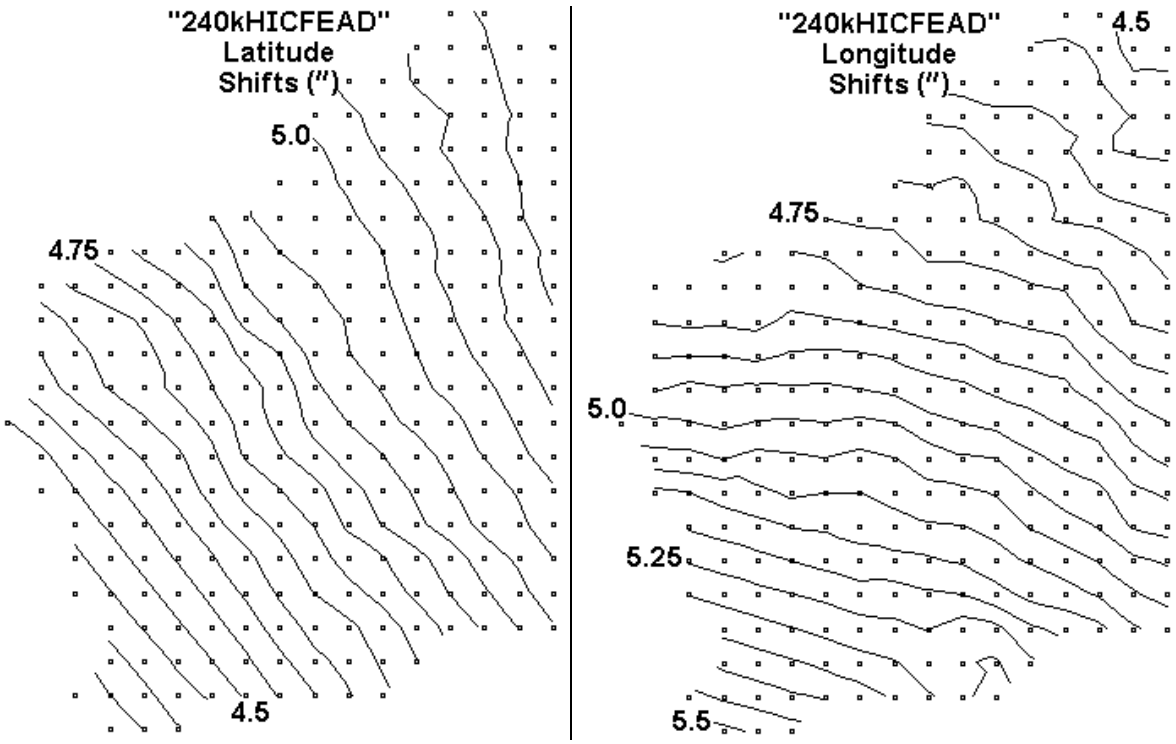


Figure I-28: Horizontal datum shifts for AGD84→GDA94 arising from HICFEAD with r_{max}=240000m.

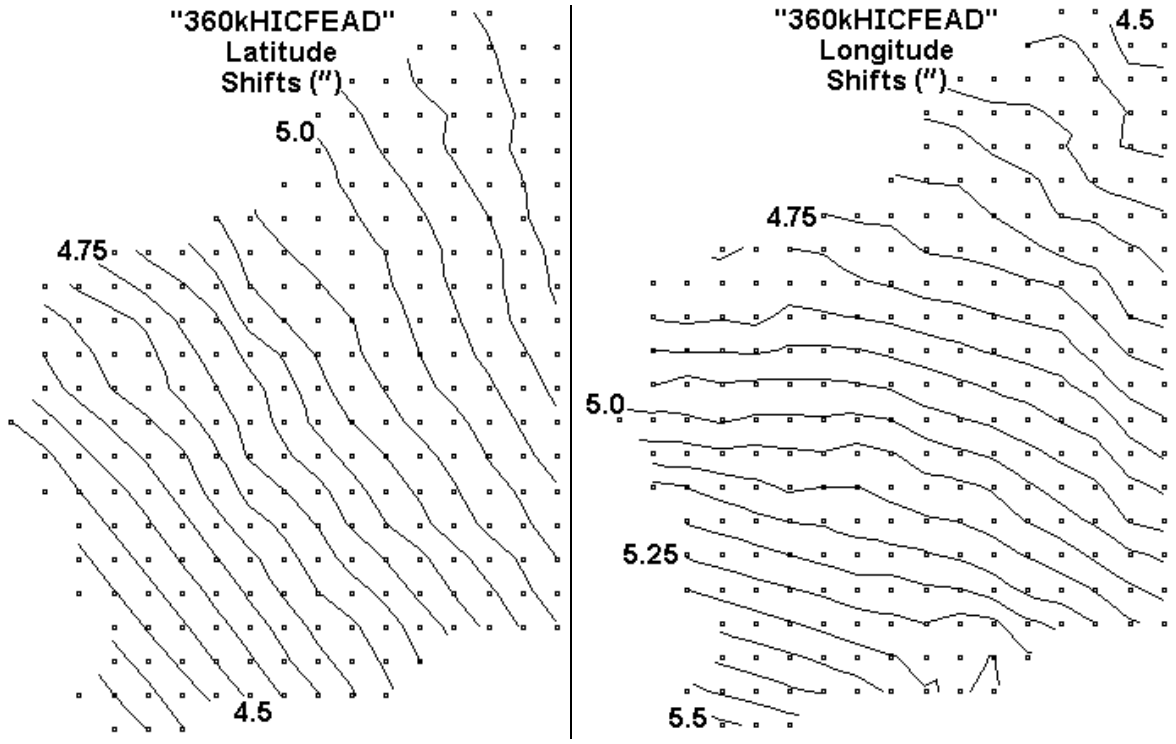


Figure I-29: Horizontal datum shifts for AGD84→GDA94 arising from HICFEAD with r_{max}=360000m.

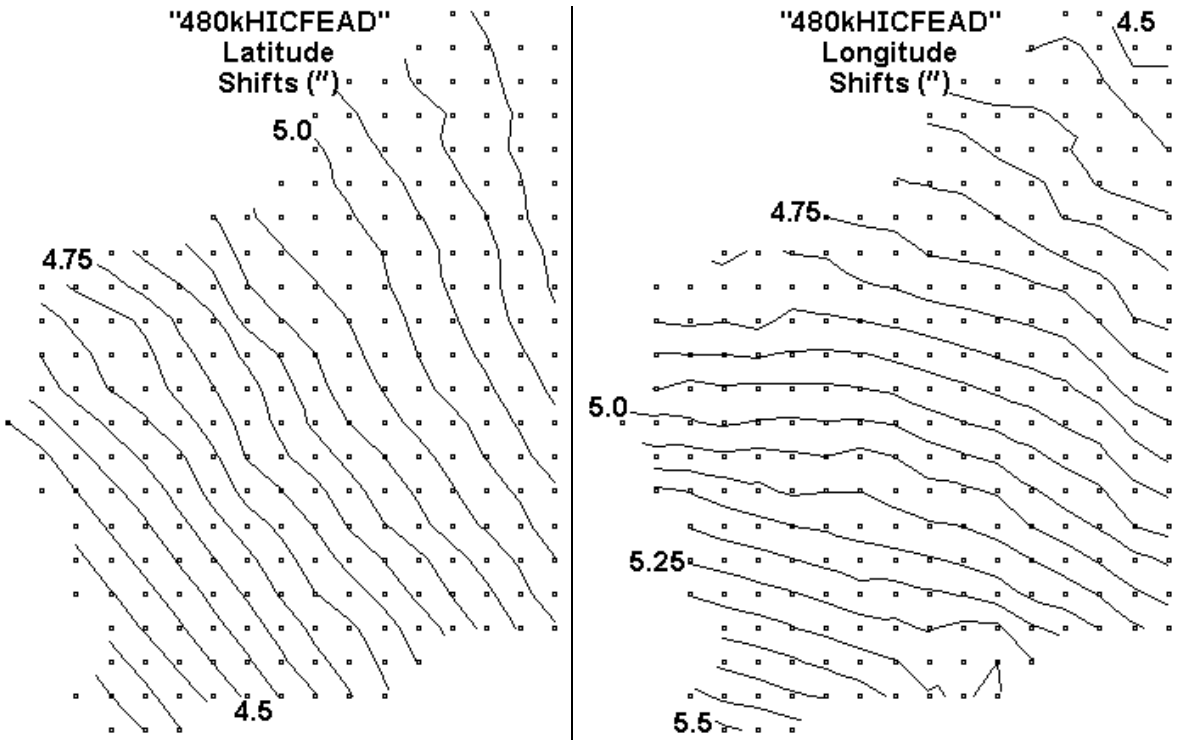


Figure I-30: Horizontal datum shifts for AGD84→GDA94 arising from HICFEAD with r_{max}=480000m.

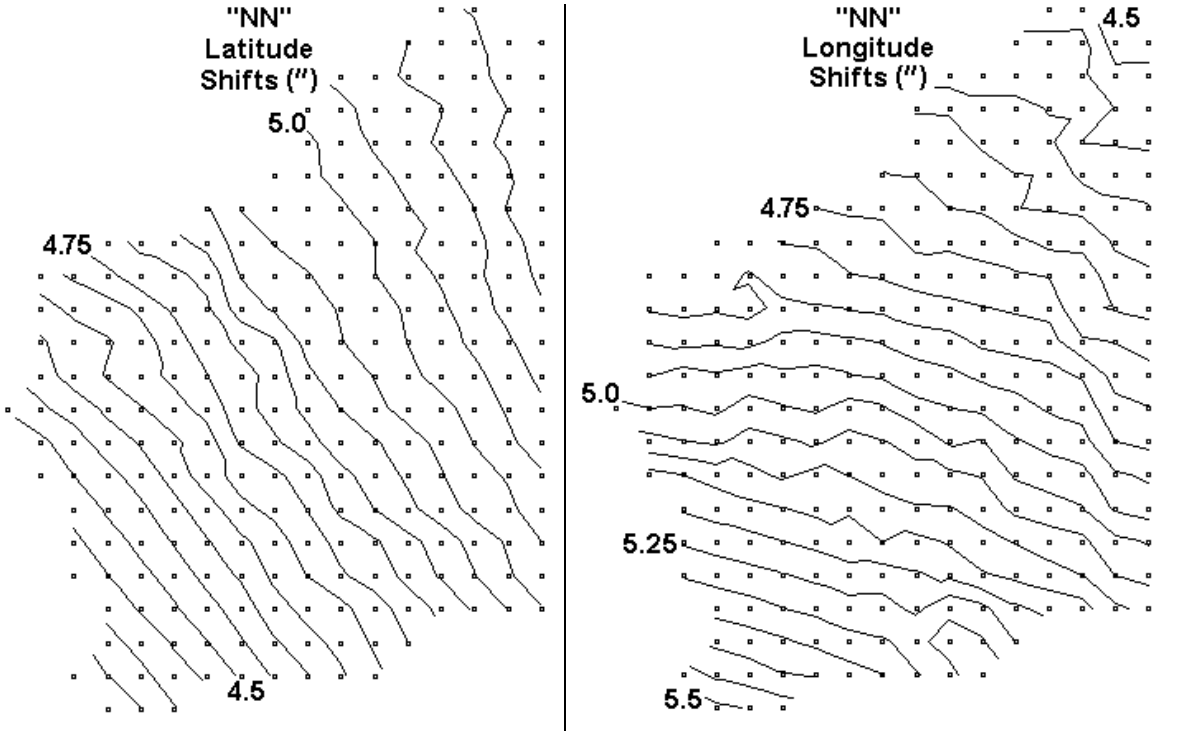


Figure I-31: Horizontal datum shifts for AGD84→GDA94 arising from Nearest Neighbour.

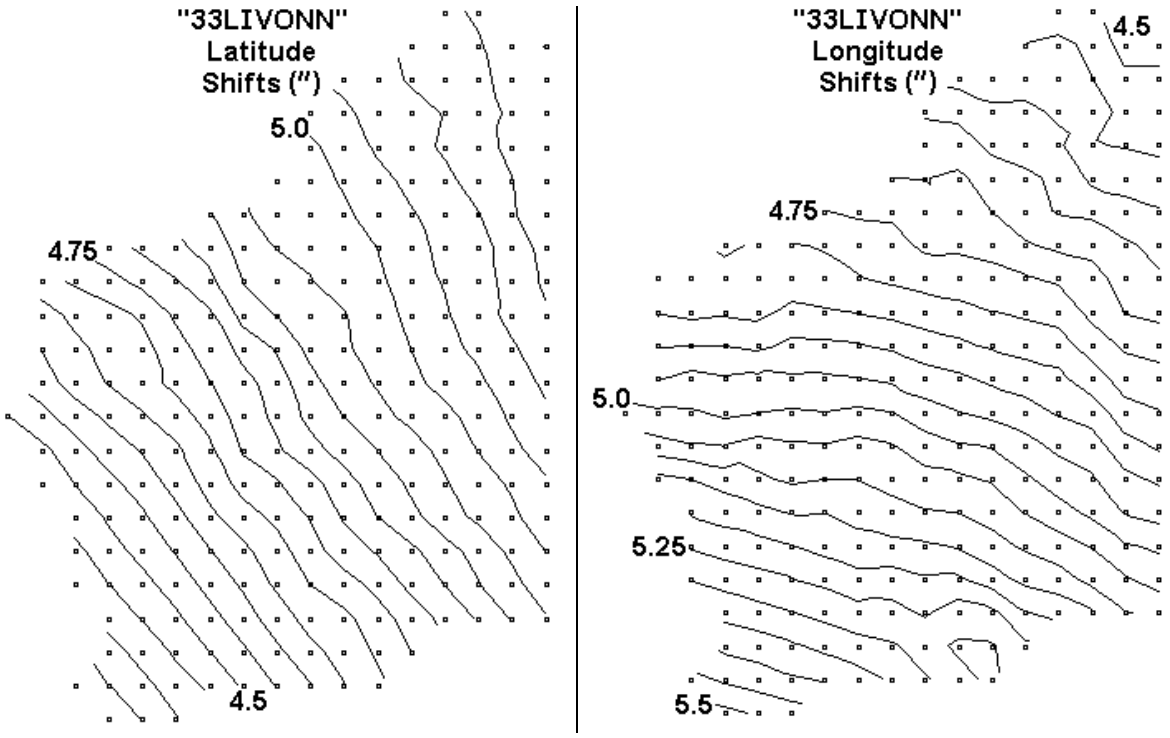


Figure I-32: Horizontal datum shifts for AGD84→GDA94 arising from LIVONN with transition interval 33.3333%.

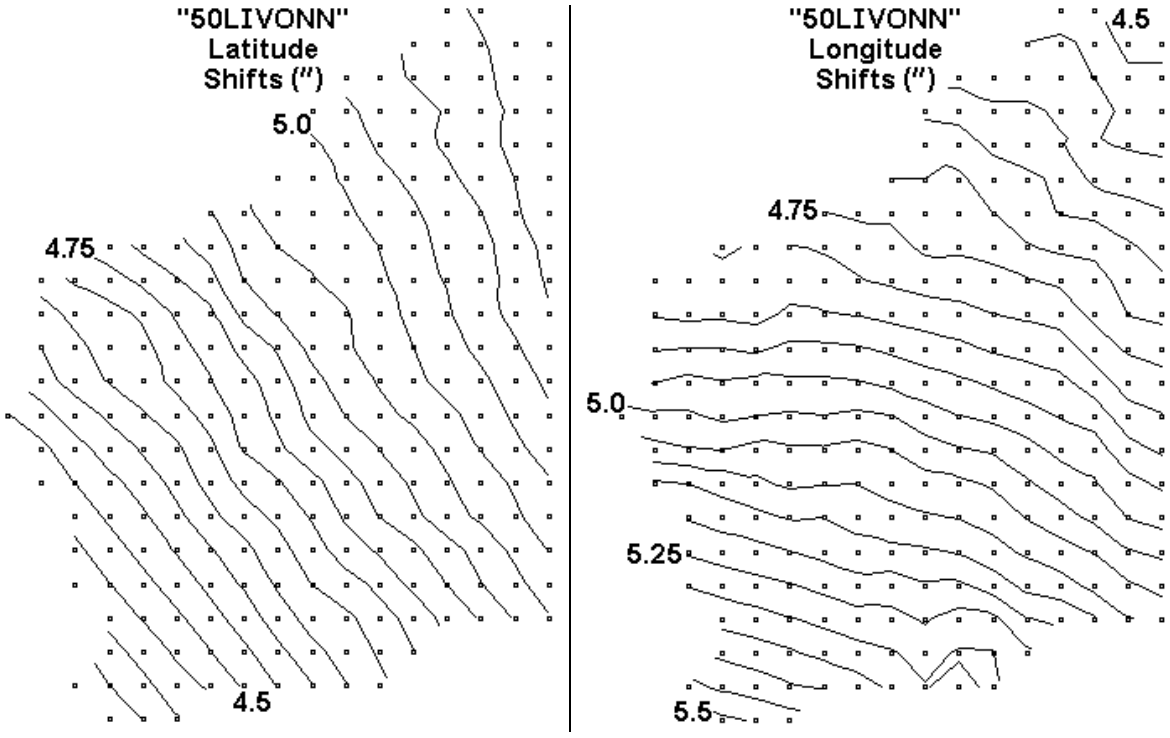


Figure I-33: Horizontal datum shifts for AGD84→GDA94 arising from LIVONN with transition interval 50%.

I.2 Slovenia (D48 to D48)

Selected contour maps arising from MREs described in Chapter 12 are listed below.

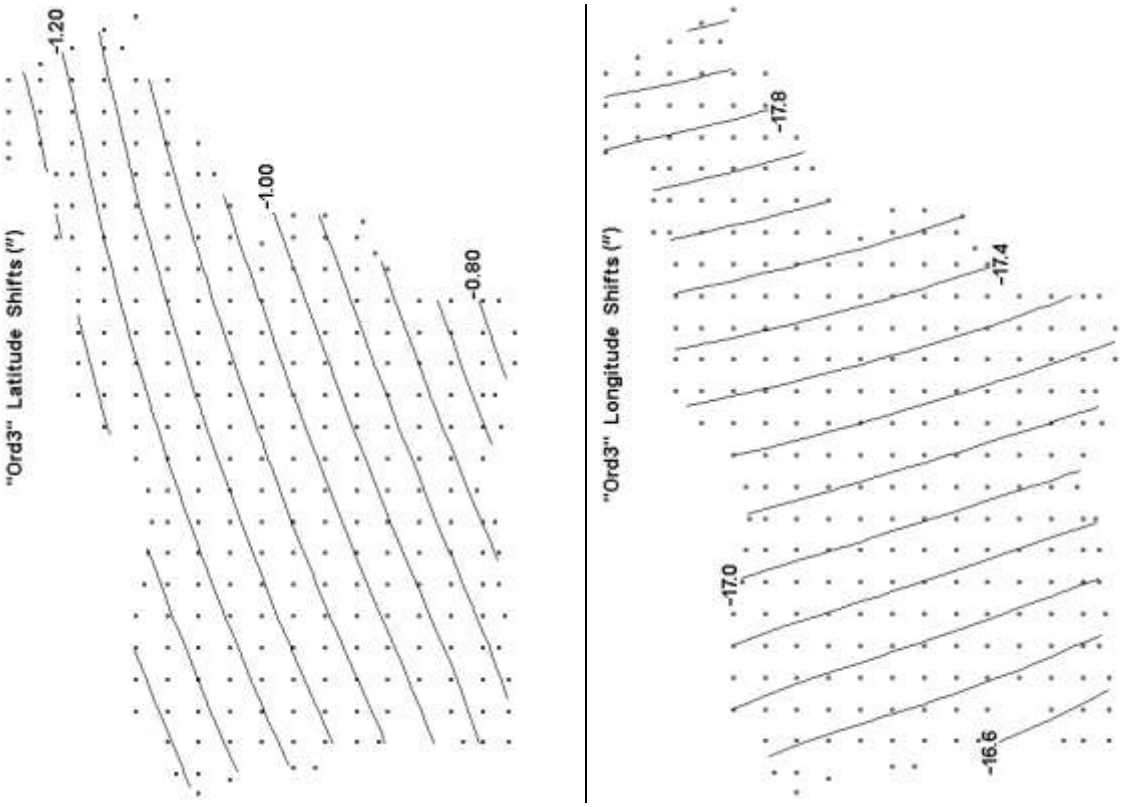


Figure I-34: Horizontal datum shifts for D48→D96 arising from the Ordinary MREs with top power 3.

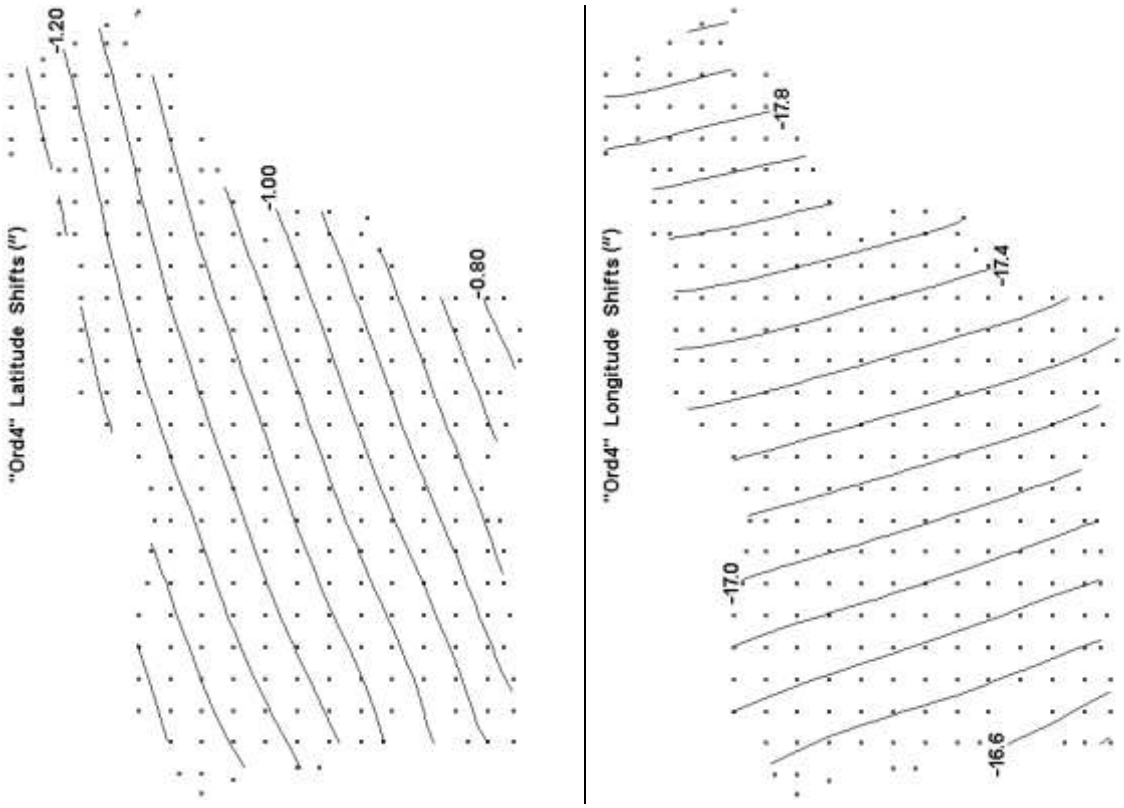


Figure I-35: Horizontal datum shifts for D48→D96 arising from the Ordinary MREs with top power 4.

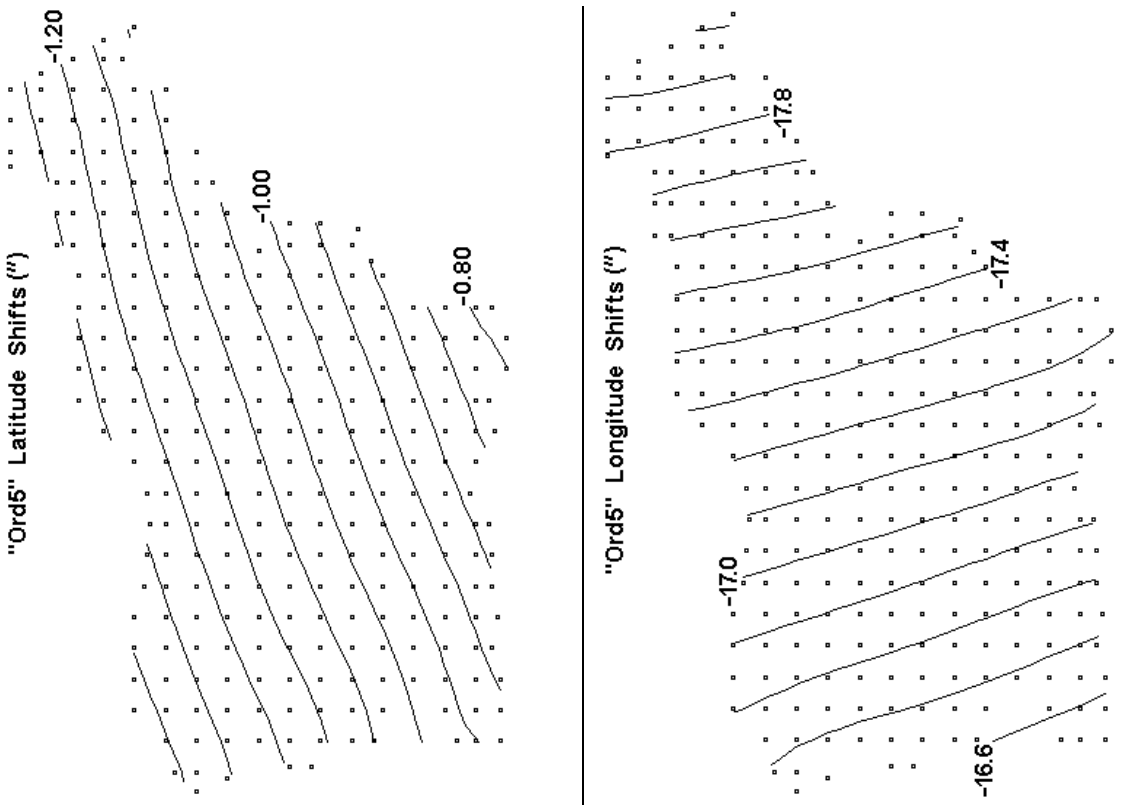


Figure I-36: Horizontal datum shifts for D48→D96 arising from the Ordinary MREs with top power 5.

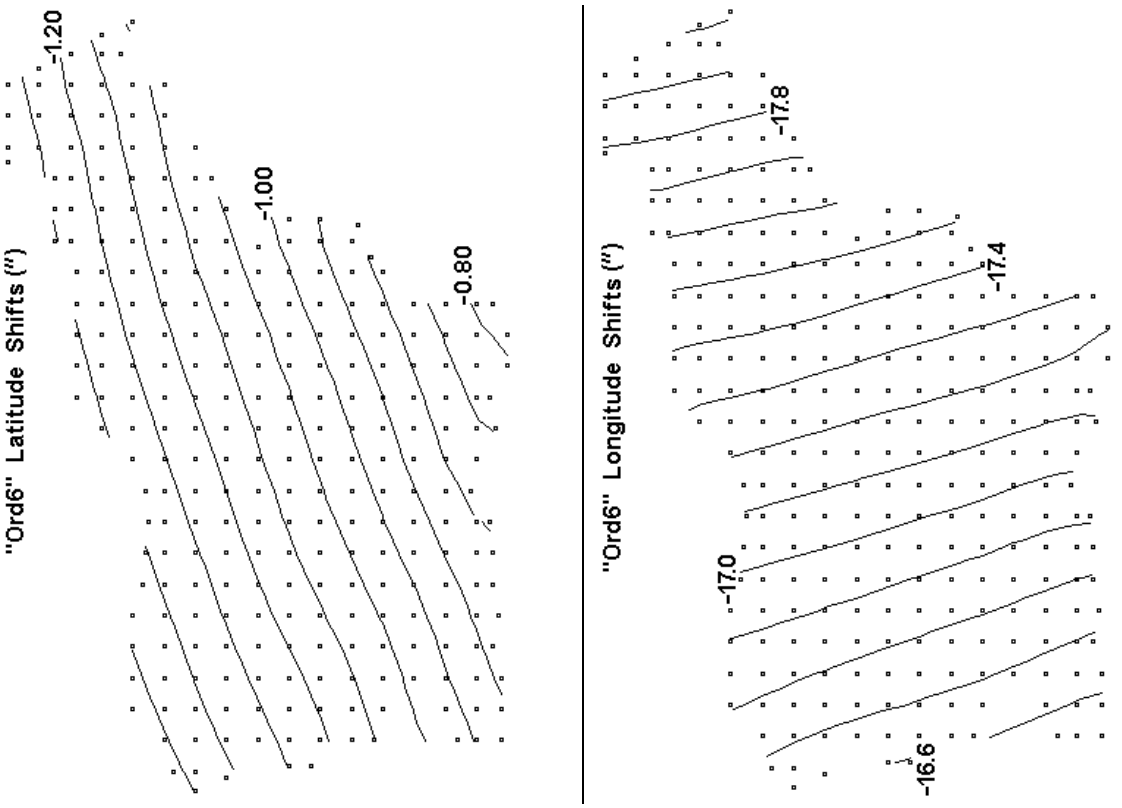


Figure I-37: Horizontal datum shifts for D48→D96 arising from the Ordinary MREs with top power 6.

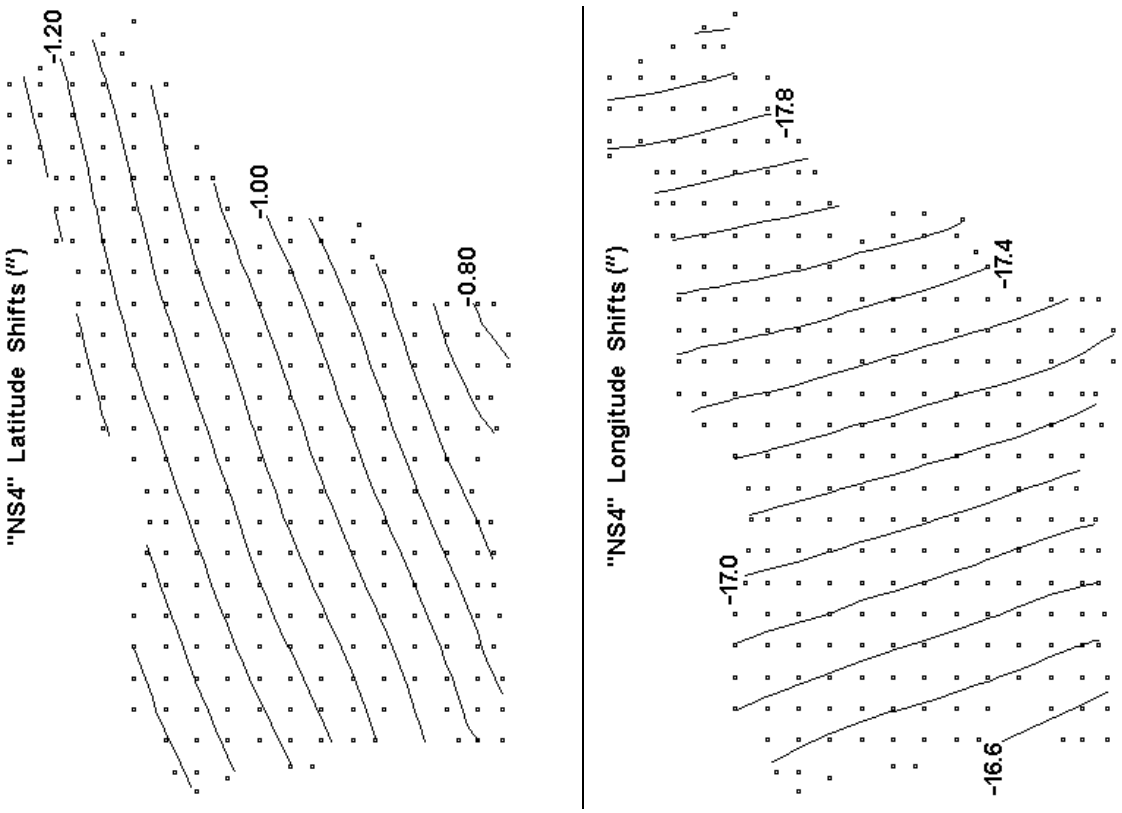


Figure I-38: Horizontal datum shifts for D48→D96 arising from the North/South MREs with top power 4.

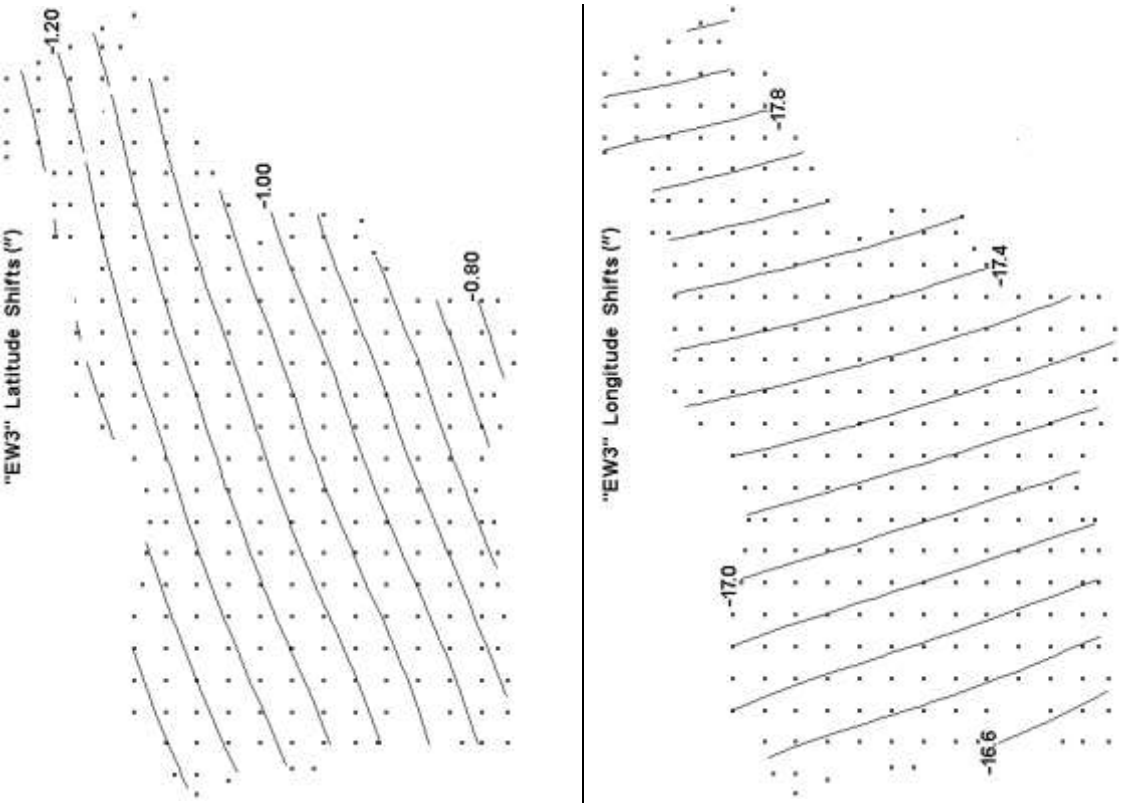


Figure I-39: Horizontal datum shifts for D48→D96 arising from the East/West MREs with top power 3.

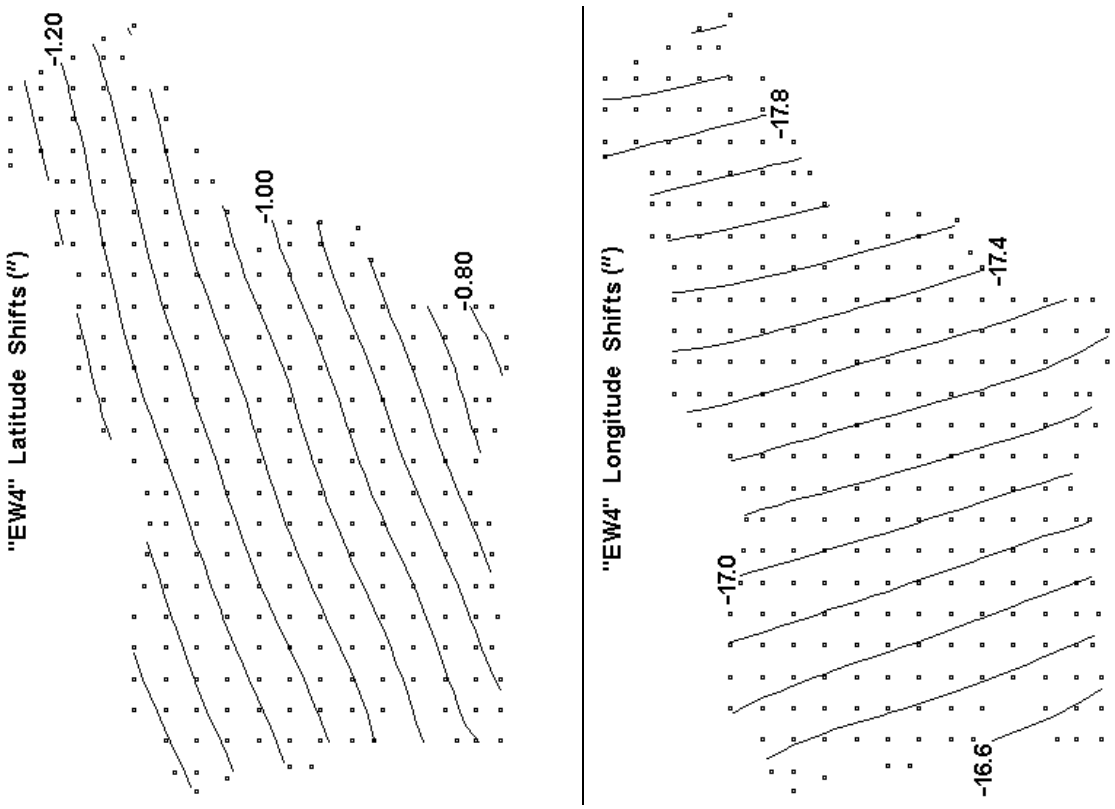


Figure I-40: Horizontal datum shifts for D48→D96 arising from the East/West MREs with top power 4.

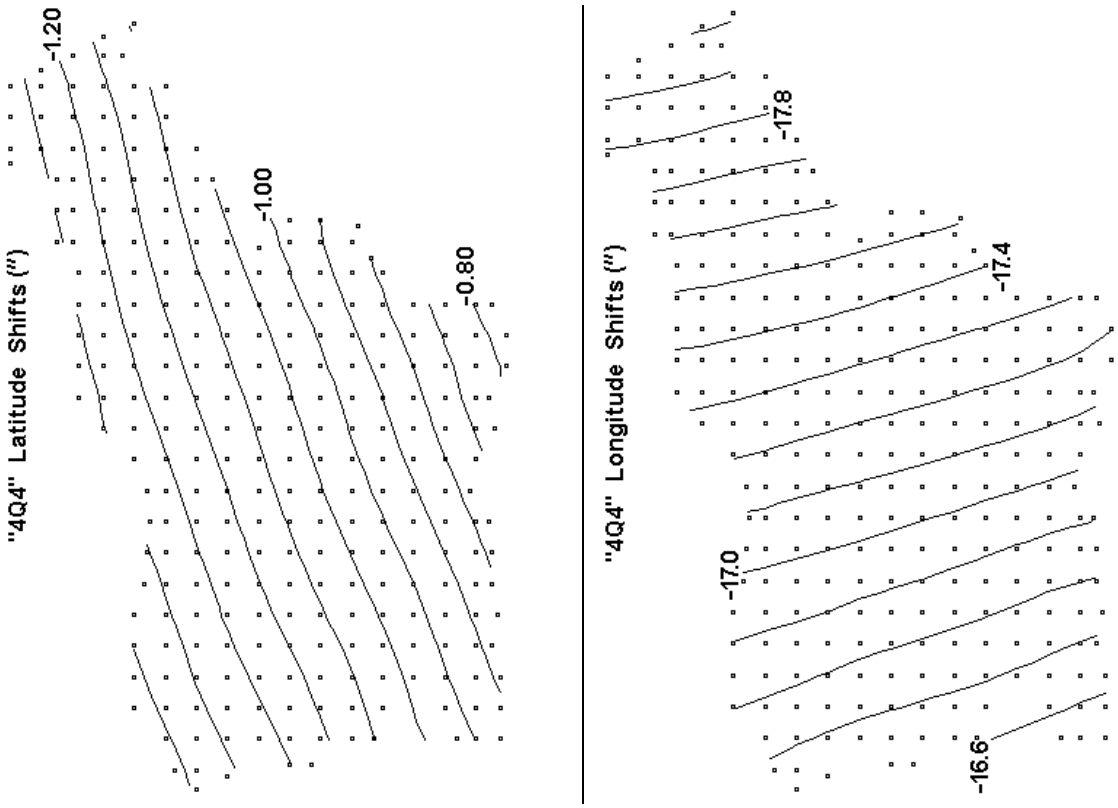


Figure I-41: Horizontal datum shifts for D48→D96 arising from the Four-Quadrant MREs with top power 4.

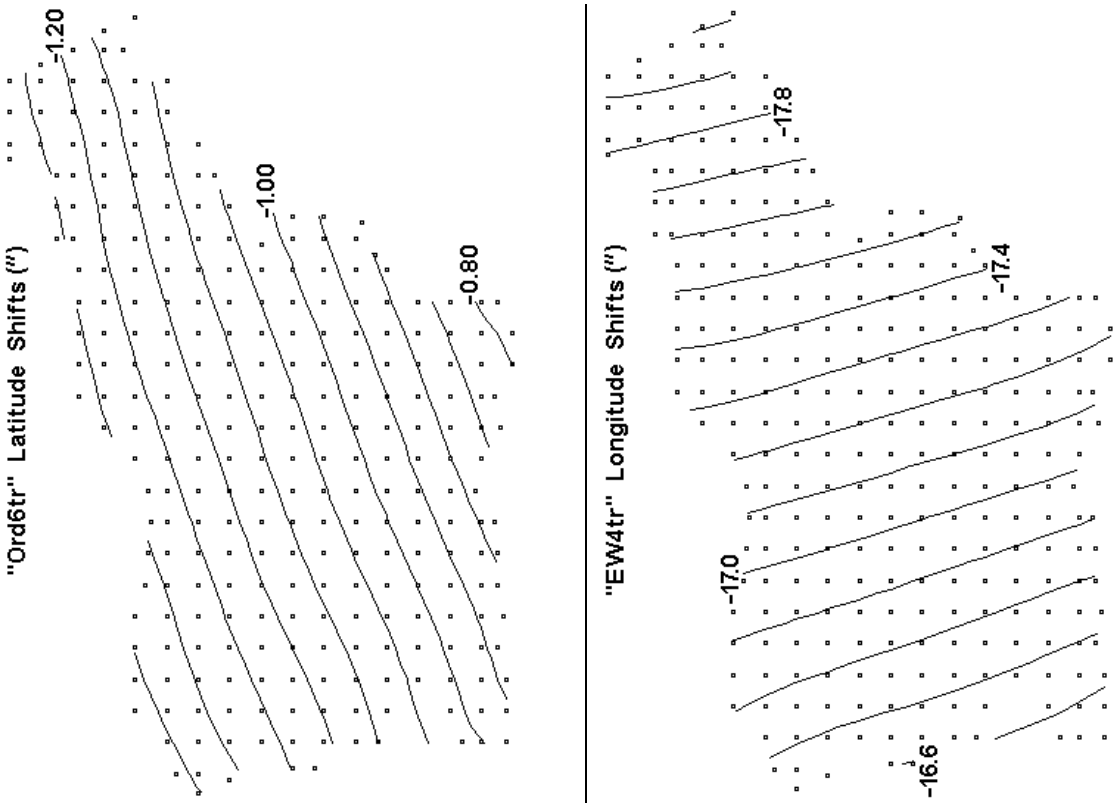


Figure I-42: Horizontal datum shifts for D48→D96 arising from the trimmed pairing of MREs Ord6tr (for latitude) and EW4tr (for longitude).

I.3 Great Britain (ETRS89 to OSGB36)

The contour maps arising from MREs described in Chapter 13 are listed below.

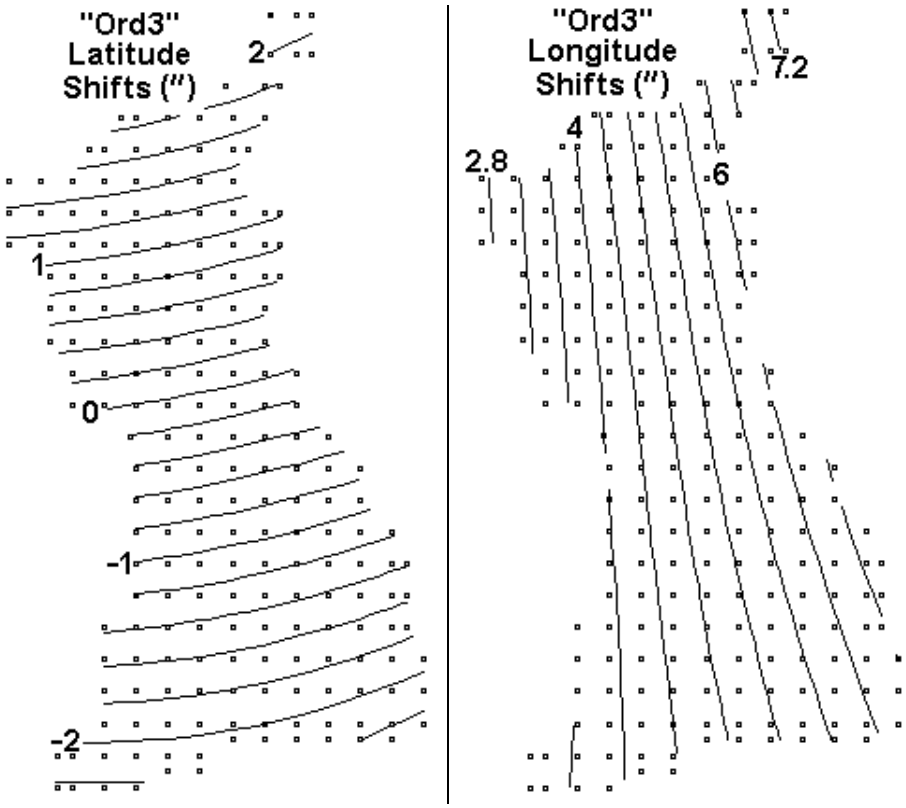


Figure I-43: Horizontal datum shifts for ETRS89→OSGB36 arising from the Ordinary MREs with top power 3.

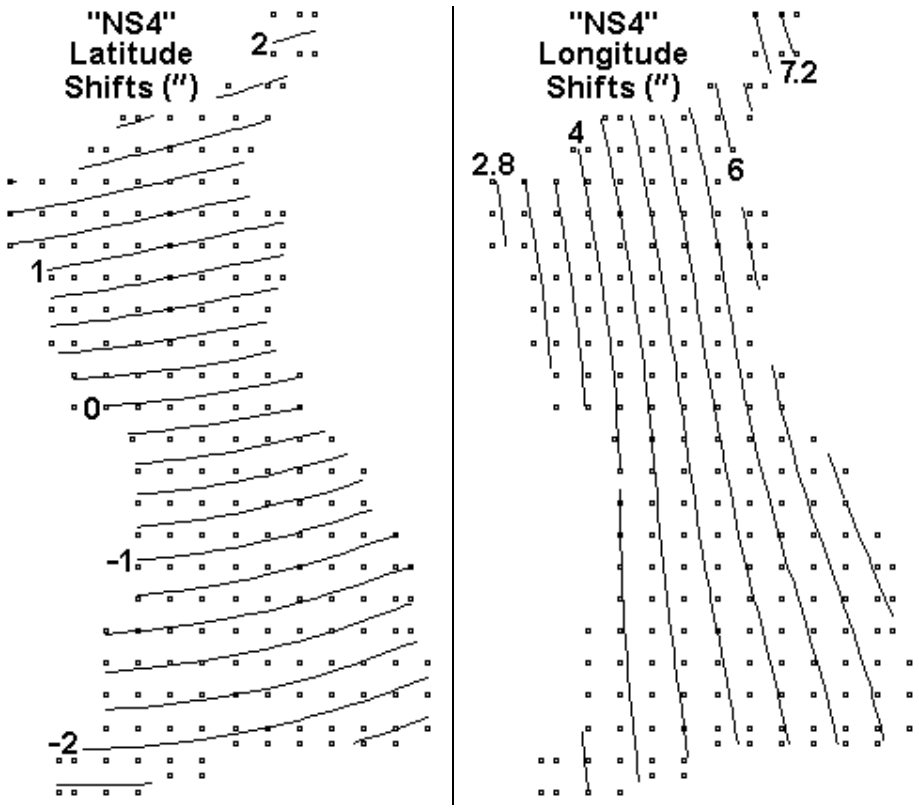


Figure I-44: Horizontal datum shifts for ETRS89→OSGB36 arising from the Ordinary MREs with top power 4.

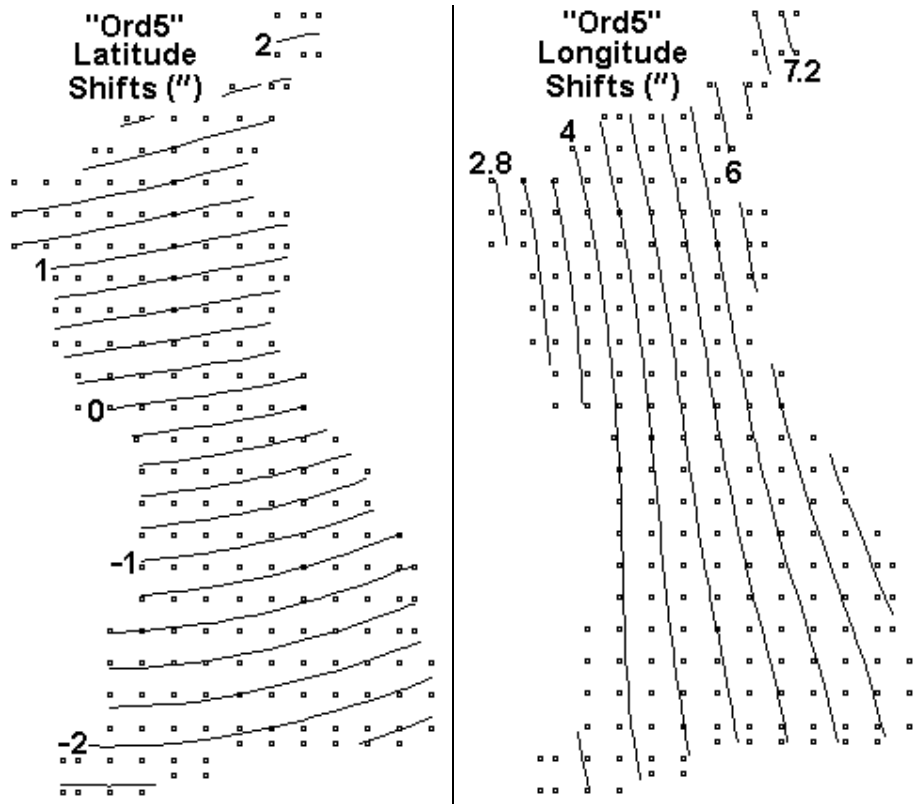


Figure I-45: Horizontal datum shifts for ETRS89→OSGB36 arising from the Ordinary MREs with top power 5.

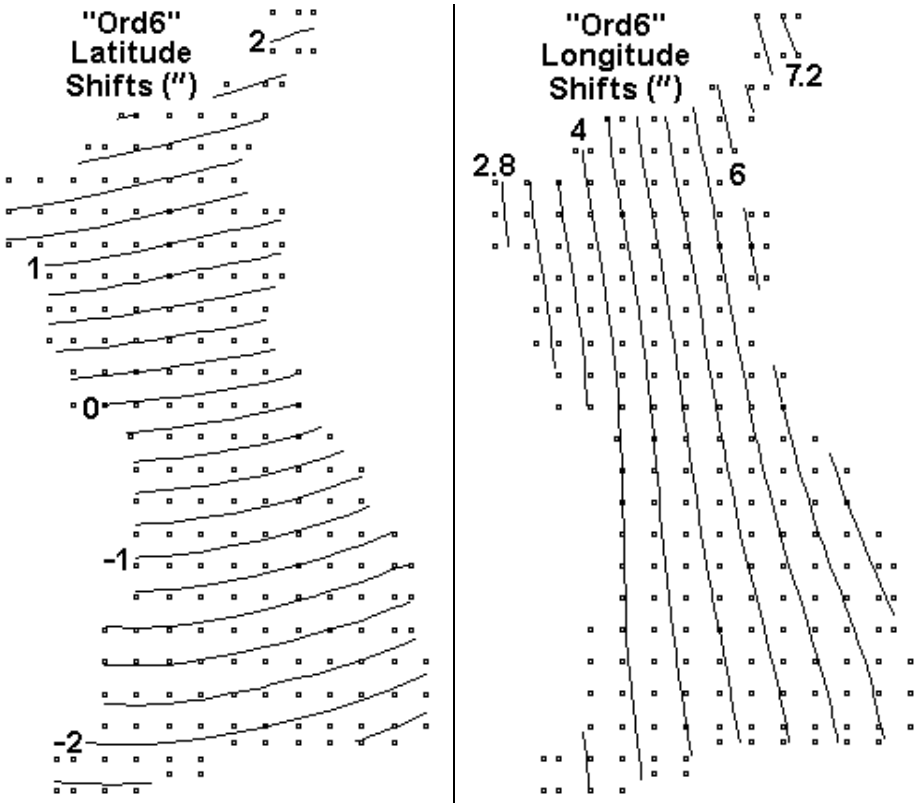


Figure I-46: Horizontal datum shifts for ETRS89→OSGB36 arising from the Ordinary MREs with top power 6.

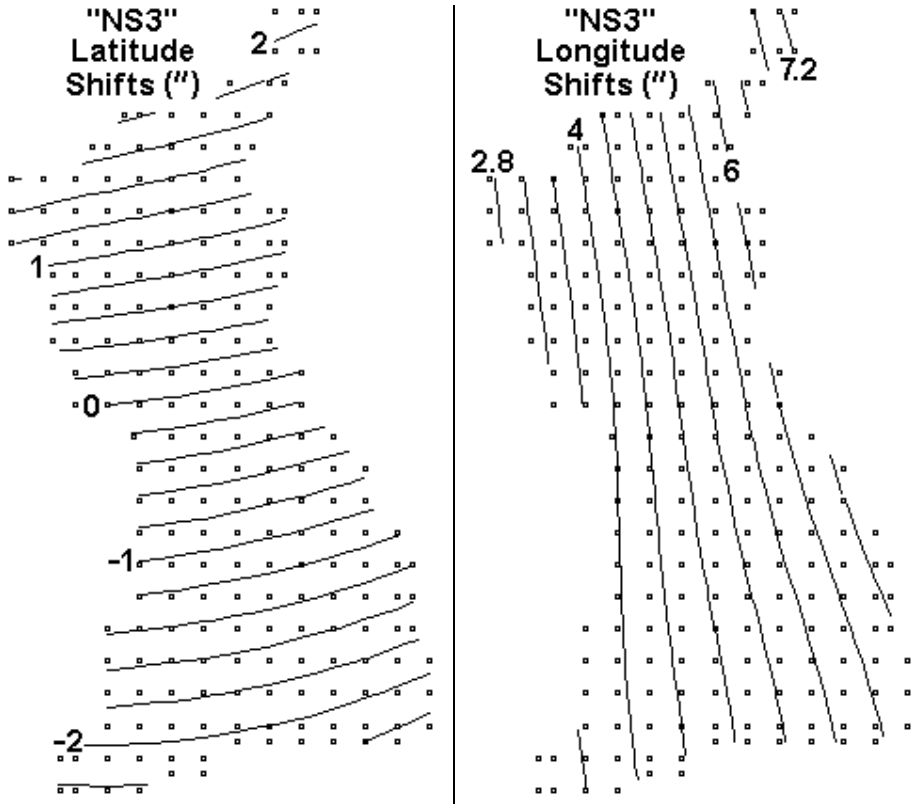


Figure I-47: Horizontal datum shifts for ETRS89→OSGB36 arising from the North/South MREs with top power 3.

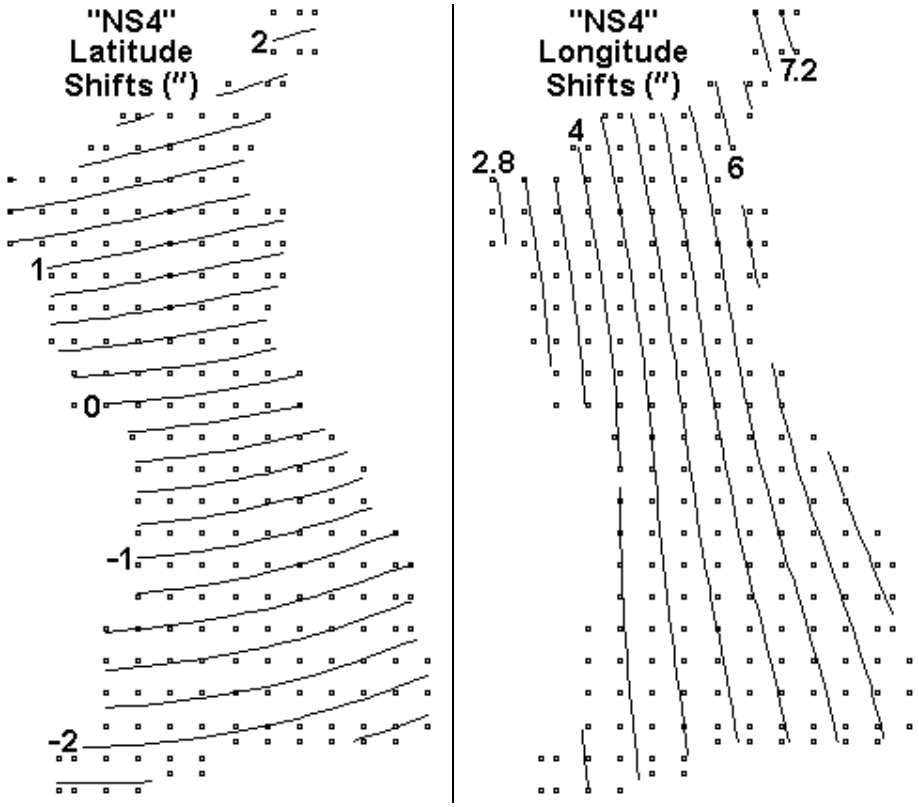


Figure I-48: Horizontal datum shifts for ETRS89→OSGB36 arising from the North/South MREs with top power 4.

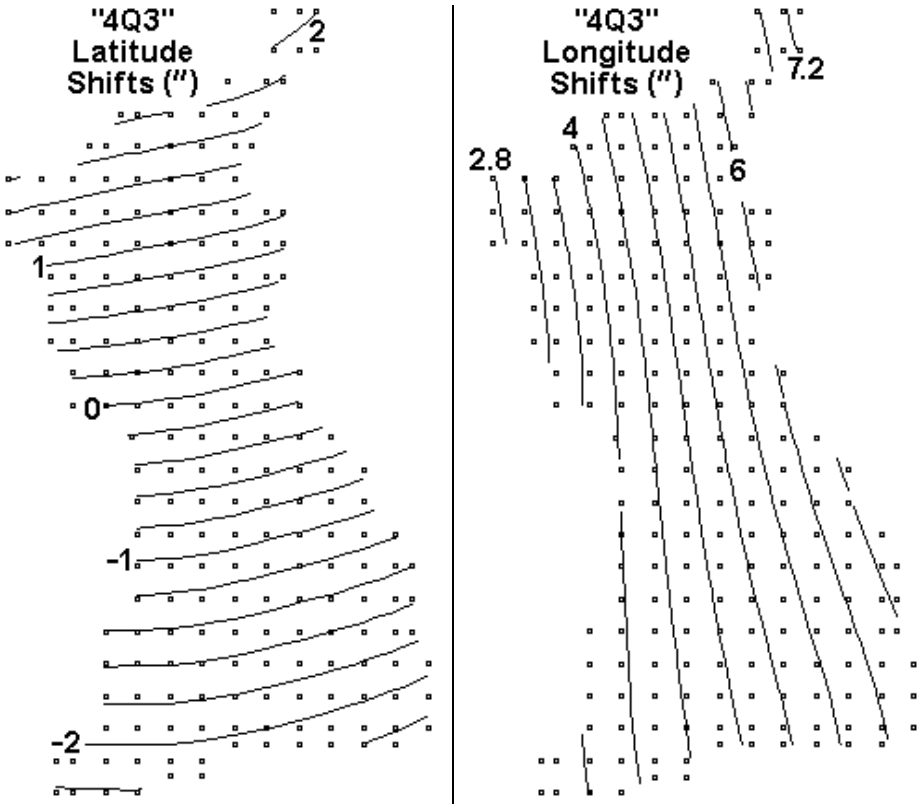


Figure I-49: Horizontal datum shifts for ETRS89→OSGB36 arising from the Four-Quadrant MREs with top power 3.

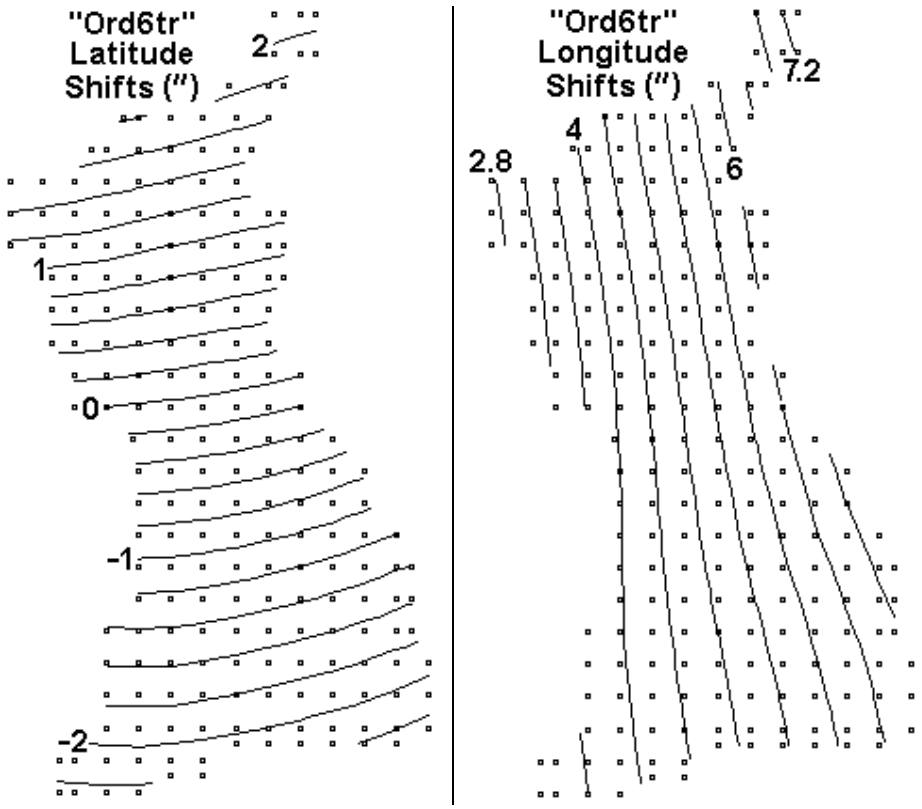


Figure I-50: Horizontal datum shifts for ETRS89→OSGB36 arising from the trimmed Ordinary MREs with top power 6.

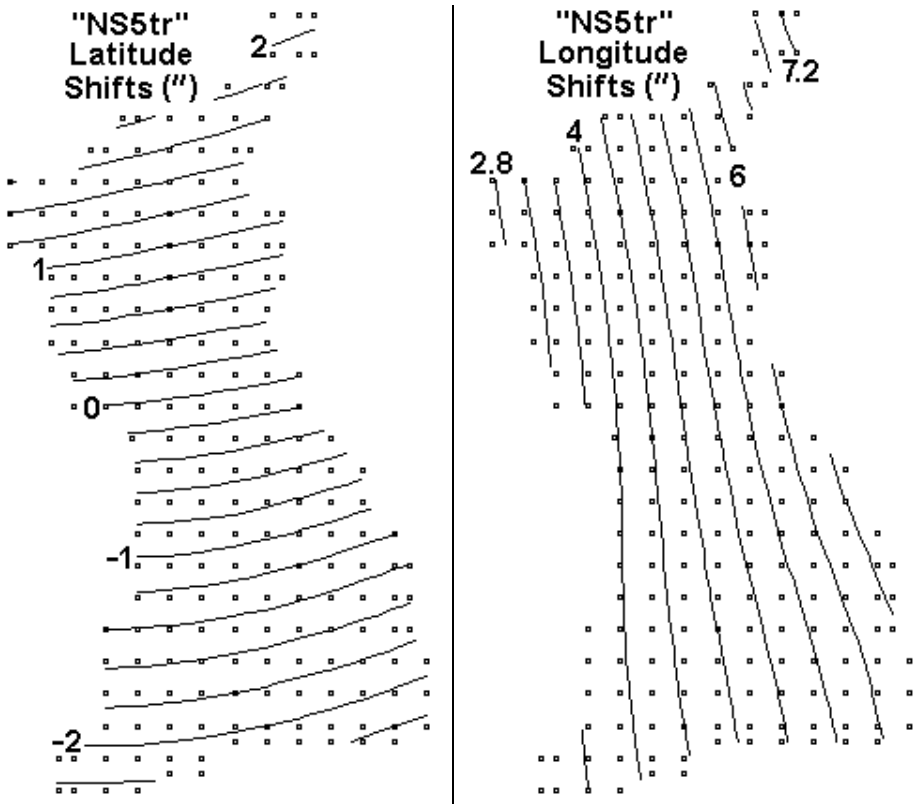


Figure I-51: Horizontal datum shifts for ETRS89→OSGB36 arising from the trimmed North/South MREs with top power 5.