Derivation of 9-parameter affine 3D geodetic datum transformations

A. C. Ruffhead

School of Architecture, Computing and Engineering, University of East London, UK.
*Corresponding author, email ruffhead40b@yahoo.co.uk

This article was published in August 2023 in *Survey Review* 55 (392) 441-450.

https://doi.org/10.1080/00396265.2022.2137094
DOI:10.1080/00396265.2022.2137094

ABSTRACT

This paper proposes a new method of deriving 9-parameter affine 3D datum transformations by ordinary least-squares. Unlike previous methods, it covers all versions of the transformation. Initially, an 'average' scale factor is computed by distance analysis. Removing the scaling effect, the 'RIGOPT' subroutine is applied to optimise the rigid transformation that consists of 3 translations and 3 rotations. Using an equivalent-enlargement hypothesis, the number of scale factors is increased to 3 by a short series of single-search-direction optimisations. The minimisation of residuals is verified by enclosing-interval analysis. The case studies cover datasets in Western Australia, Great Britain and Sweden.

Keywords: datum transformations, geodetic datums, affine transformations, non-linear optimisation, equivalent-enlargement hypothesis

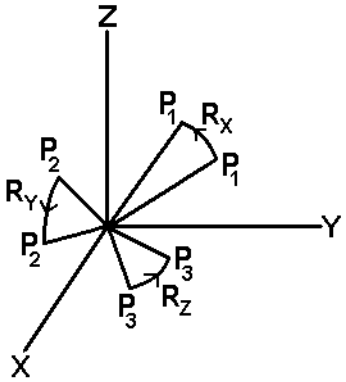
1. Introduction

As noted by Paláncz *et al.* (2008), because of distortions between the traditional terrestrial and the GPS-derived networks, '7-parameter similarity transformations in some cases may not offer satisfactory precision'. The similarity transformation of GPS local coordinates to the local Hungarian system is quoted as an example. The 9-parameter affine transformation with 3 scale factors is described as 'not only a logical extension but even a generalisation of the 7-parameter similarity transformation model'.

The 9-parameter 3D affine transformation is generally applied to geocentric Cartesian coordinates: (X_s, Y_s, Z_s) in the source datum is transformed to (X_t, Y_t, Z_t) in the target datum. Where there are more than 3 points known in both systems, the coordinates are computed by a form of least-squares optimisation. In this paper, as in those listed under 'Existing methods of derivation', that form is ordinary least-squares, which means the observations are preserved with no attempt to 'adjust' them.

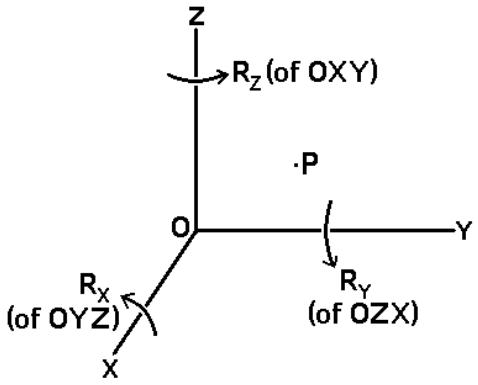
The parameters used are:

- Translation parameters ΔX, ΔY, ΔZ.
- Scaling parameters. These can take the form of scale factors, S_X, S_Y and S_Z, which are in the directions of the axes OX, OY and OZ. They can also take the form of scale changes ΔS_X, ΔS_Y and ΔS_Z which are often expressed in parts per million (ppm). Scale factors and scale changes are inter-related by the identities $S_X = 1 + \Delta S_X$, etc.
- Rotation parameters R_X, R_Y, R_Z. In this paper they describe rotation of position vectors about Cartesian axes, as illustrated in Fig. 1. In Europe it is the more commonly used convention. It is used by the International Association of Geodesy and is recommended by ISO (2007). One characteristic is that a positive rotation about the Z-axis has the effect of increasing longitude.



1 Position vector (PV) rotations

Some authors prefer the coordinate-frame rotation convention shown in Fig. 2 which has the opposite effect to those in Fig. 1. Coordinate-frame rotation parameters are therefore opposite in sign to the position-vector rotations used here.



2 Coordinate frame (CF) rotations

Theoretically, there are 6 ways of applying the rotations R_X, R_Y, R_Z, depending on the order in which they are applied. In practice, R_Y is always the second to be applied. That means the rotation matrix **R** takes one of the following forms, where $\cos R_X$ is denoted by c_X, $\sin R_X$ by s_X, etc.

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

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

Each of these equations can be expressed in terms of a single matrix.

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

$$\mathbf{R}_{XYZ} = \begin{bmatrix} c_Y c_Z & -c_Y s_Z & s_Y \\ s_X s_Y c_Z + c_X s_Z & c_X c_Z - s_X s_Y s_Z & -s_X c_Y \\ s_X s_Z - c_X s_Y c_Z & c_X s_Y s_Z + s_X c_Z & c_X c_Y \end{bmatrix} \tag{4}$$

The sine terms in equations (1) to (4) change sign if the rotation convention is based on coordinate frame rather than position vector. There is a similar knock-on effect in Appendix A.

For the purposes of this paper, the rotation matrix **R** is assumed to be $\mathbf{R}_{ZYX}$, meaning that the rotation R_X is effectively applied first even if (3) is substituted for (2).

The use of 3 scale factors implies there is some reason to expect differences in scale in the directions of the axes OX, OY and OZ. This might be the case if either or both of the datasets was obtained by a 2+1D process, particularly if the local vertical direction is close to being in one of the planes OXY, OYZ and OZX.

When there are 3 scale factors, the scaling matrix **S** is given by

$$\mathbf{S} = \begin{bmatrix} S_X & 0 & 0 \\ 0 & S_Y & 0 \\ 0 & 0 & S_Z \end{bmatrix} \tag{5}$$

There are two ways in which the scaling matrix **S** can be applied. One is the 'SR' permutation in which **R** is applied before **S**, favoured by Han (2010), Paláncz *et al.* (2008), Späth (2004) and Varga *et al.* (2017). The SR affine transformation formula is as follows:

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{SR} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} \quad (6)$$

The other way is the 'RS' permutation in which **S** is applied before **R**, favoured by Andrei (2006) and Fan (2009). The SR affine transformation formula is as follows:

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{RS} \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} \quad (7)$$

An equivalent form of the RS affine formula is

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{R} \begin{bmatrix} S_X X_s \\ S_Y Y_s \\ S_Z Z_s \end{bmatrix} \quad (8)$$

There is a little-used 8-parameter affine transformation in which two of the scale factors are combined into one that applies to a whole plane. Andrei (2006) explored it for local level (as opposed to geocentric) coordinates. His rationale was that one scale factor would apply to horizontal coordinates and the other to heights. The use of an 8-parameter affine transformation on geocentric affine coordinates is not advocated in this paper. However, computation of 8 parameters will be considered as an intermediate stage in the derivation of the full 9 parameters.

2. Existing methods of derivation

Whether the 9-parameter affine model uses a multiplicative matrix of the form **RS** or **SR**, it is non-linear with respect to the rotation and scaling parameters. This complicates the process of optimising the parameters by least-squares. 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 is an iterative process that requires initial approximations of the rotations, but nothing is said about how they are obtained. The numerical examples involve rotations in the range 28° to 258°, so are far removed from geodetic datums. The examples require 42 and 137 iterations.
- Watson (2006) describes a method of obtaining the SR version from a dataset of *N* control points. It is a faster iterative process than that of Späth. The numerical examples are those of Späth (2004), which are far removed from geodetic datums.
- 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.

Only 5 of these methods cover the case of *N* control points. None of them considers more than one version of the 9-parameter affine transformation. Furthermore, in the author's view, all of them use concepts that are unnecessarily complicated, with the possible exception of Andrei's 'least square adjustment method' which is insufficiently documented.

3. New method of derivation

The new method of deriving the optimal 9-parameter affine transformation has four stages, described in more detail in subsections 3.1 to 3.4. Each stage uses the subroutine RIGOPT described in Appendix A.

- Seven-parameter optimisation: the method given in Ruffhead (2021) is applied to obtain the best-fitting 7-parameter conformal (Helmert) model.
- Eight-from-seven optimisation: an 8-parameter affine transformation is obtained by variations from the best-fitting Helmert model, treating two of the scale factors as equal.
- Nine-from-eight optimisation: 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).
- Scale ratios correction: 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 notation is introduced.

- $(X_{s,m}, Y_{s,m}, Z_{s,m})$ denotes the central point in the source datum where $X_{s,m} = (1/n)\sum_{i=1}^{n} X_{s,i}$, etc.
- $(X_{t,m}, Y_{t,m}, Z_{t,m})$ denotes the central point in the target datum where $X_{t,m} = (1/n)\sum_{i=1}^{n} X_{t,i}$, etc.
- $d_{s,i}$ denotes the distance from data point $(X_{s,i}, Y_{s,i}, Z_{s,i})$ to the central point in the source datum.
- $d_{t,i}$ denotes the distance from data point $(X_{t,i}, Y_{t,i}, Z_{t,i})$ to the central point in the target datum.
- d_X, d_Y and d_Z denote distance components $|X - X_{s,m}|$, $|Y - Y_{s,m}|$ and $|Z - Z_{s,m}|$ for use in summations based on the data points.
- d_{XZ}, denotes distance between data points in the XZ plane with respect to the source datum, so that $d_{XZ}^2 = d_X^2 + d_Z^2$.
- S_{DA} denotes the scale factor in the optimal Helmert transformation. Ruffhead (2021) showed that this is derived from distance analysis, hence the 'DA' subscript.
- RMSD is the root-mean-square distance of the residuals between the original target coordinates and the computed coordinates obtained from the latest affine model. This, or equivalently its square, is the quantity to be minimised under ordinary least squares.

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 Helmert model.

$$\sum(S_X d_X)^2 + \sum(S_Y d_Y)^2 + \sum(S_Z d_Z)^2 = \sum\left(S_{\mathrm{DA}}\sqrt{d_X^2 + d_Y^2 + d_Z^2}\right)^2 \tag{9}$$

The equivalent-enlargement hypothesis can also be written as

$$S_X^2\sum d_X^2 + S_Y^2\sum d_Y^2 + S_Z^2\sum d_Z^2 = S_{\mathrm{DA}}^2\sum(d_X^2 + d_Y^2 + d_Z^2) \tag{10}$$

It should be noted that the right-hand side of this equation is constant.

3.1. Seven-parameter optimisation

This stage is based on the Helmert optimisation method in Ruffhead (2021), which shows that the scale factor of the least-squares solution is

$$S_{\mathrm{DA}} = \sum_{i=1}^{n} d_{s,i}d_{t,i} \;/\; \sum_{i=1}^{n} d_{s,i}^2 \tag{11}$$

The subroutine RIGOPT is applied with input arguments $n, \{S_{\mathrm{DA}}X_{s,i}\}, \{S_{\mathrm{DA}}Y_{s,i}\}, \{S_{\mathrm{DA}}Z_{s,i}\}, \{X_{t,i}\}, \{Y_{t,i}\}, \{Z_{t,i}\}$. The first 6 output arguments are ΔX, ΔY, ΔZ, R_X, R_Y, R_Z which complete the set of optimal Helmert parameters. RMSD is easily computed from the residuals.

3.2. Eight-from-seven optimisation

The RMSD from 3.1 is the minimum for the 9-parameter affine transformation subject to the constraint $S_X = S_Y = S_Z$. Treating only two of the scale factors as equal is the first step in removing that constraint. Without loss of generality, it is assumed that $S_X = S_Z = S_{XZ}$ where S_{XZ} denotes scale in the *XZ* plane.

This enables condition (10) to be rewritten

$$S_Y^2\sum d_Y^2 + S_{XZ}^2\sum d_{XZ}^2 = S_{\mathrm{DA}}^2\sum(d_Y^2 + d_{XZ}^2) \tag{12}$$

As a result, S_{XZ} can be expressed in terms of S_Y:

$$S_{XZ} = \sqrt{\frac{S_{\mathrm{DA}}^2\sum(d_Y^2+d_{XZ}^2) - S_Y^2\sum d_Y^2}{\sum d_{XZ}^2}} \tag{13}$$

This substitution is applied to each trial value of S_y.

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:

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{R}\begin{bmatrix} S_{XZ}X_s \\ S_Y Y_s \\ S_{XZ}Z_s \end{bmatrix} \tag{14}$$

The input arguments to RIGOPT for a trial value of S_Y would be n, $\{S_{XZ}X_{s,i}\}$, $\{S_Y Y_{s,i}\}$, $\{S_{XZ}Z_{s,i}\}$, $\{X_{t,i}\}$, $\{Y_{t,i}\}$, $\{Z_{t,i}\}$. The first 6 output arguments are ΔX, ΔY, ΔZ, R_X, R_Y, R_Z which complete the set of affine parameters. RMSD is easily computed from the residuals.

If the affine model is of type SR, the 6-parameter rigid transformation is treated as that which transforms the source coordinates to the 'de-scaled' target coordinates:

$$\begin{bmatrix} X_t/S_{XZ} \\ Y_t/S_Y \\ Z_t/S_{XZ} \end{bmatrix} = \begin{bmatrix} \Delta X/S_{XZ} \\ \Delta Y/S_Y \\ \Delta Z/S_{XZ} \end{bmatrix} + \mathbf{R}\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} \tag{15}$$

The input arguments to RIGOPT for a trial value of S_Y would be $n, \{X_{s,i}\}, \{Y_{s,i}\}, \{Z_{s,i}\}, \{X_{t,i}/S_{XZ}\}, \{Y_{t,i}/S_Y\}, \{Z_{t,i}/S_{XZ}\}$. The first 6 output arguments are $\Delta X/S_{XZ}$, $\Delta Y/S_Y$, $\Delta Z/S_{XZ}$, R_X, R_Y, R_Z which complete the set of affine parameters after the scale factors convert the shifts to ΔX, ΔY, ΔZ. The residuals are $\{v_{X,i}/S_{XZ}\}$, $\{v_{Y,i}/S_Y\}$, $\{v_{Z,i}/S_{XZ}\}$ which also need multiplication by the scale factors to give the true value of RMSD.

In either case (RS or SR), RMSD is minimised by treating it as a function of S_Y and applying a one-dimensional optimisation method. The starting value of S_Y is S_{DA} for which the RMS was computed in 3.1. The value of S_Y that minimises the RMS is denoted $S_{Y,prov}$.

3.3. Nine-from-eight optimisation

The final RMSD from 3.2 is the minimum for the 9-parameter affine transformation subject to the constraint $S_X = S_Y$. The next stage is removal of that constraint, while requiring that S_Y is fixed, *ie* $S_Y = S_{Y,prov}$.

This enables condition (10) to be rewritten

$$S_X^2\sum d_X^2 + S_{Y,prov}^2\sum d_Y^2 + S_Z^2\sum d_Z^2 = S_{\mathrm{DA}}^2\sum(d_X^2 + d_Y^2 + d_Z^2) \tag{16}$$

As a result, S_Z can be expressed in terms of S_X:

$$S_Z = \sqrt{\frac{S_{\mathrm{DA}}^2\sum(d_X^2+d_Y^2+d_Z^2) - S_{Y,prov}^2\sum d_Y^2 - S_X^2\sum d_X^2}{\sum d_Z^2}} \tag{17}$$

This substitution is applied to each trial value of S_X.

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:

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{R}\begin{bmatrix} S_X X_s \\ S_{Y,prov} Y_s \\ S_Z Z_s \end{bmatrix} \tag{18}$$

The input arguments to RIGOPT for a trial value of S_x would be n, $\{S_X X_{s,i}\}$, $\{S_{Y,prov} Y_{s,i}\}$, $\{S_X Z_{s,i}\}$, $\{X_{t,i}\}$, $\{Y_{t,i}\}$, $\{Z_{t,i}\}$. The first 6 output arguments are ΔX, ΔY, ΔZ, R_X, R_Y, R_Z which complete the set of affine parameters. RMSD is easily computed from the residuals.

If the affine model is of type SR, the 6-parameter rigid transformation is treated as that which transforms the source coordinates to the 'de-scaled' target coordinates:

$$\begin{bmatrix} X_t/S_X \\ Y_t/S_{Y,prov} \\ Z_t/S_Z \end{bmatrix} = \begin{bmatrix} \Delta X/S_X \\ \Delta Y/S_{Y,prov} \\ \Delta Z/S_Z \end{bmatrix} + \mathbf{R}\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} \tag{19}$$

The input arguments to RIGOPT for a trial value of S_x would be $n, \{X_{s,i}\}, \{Y_{s,i}\}, \{Z_{s,i}\}, \{X_{t,i}/S_X\}, \{Y_{t,i}/S_{Y,prov}\}, \{Z_{t,i}/S_Z\}$. The first 6 output arguments are $\Delta X/S_X$, $\Delta Y/S_{Y,prov}$, $\Delta Z/S_Z$, R_X, R_Y, R_Z which complete the set of affine parameters after the scale factors convert the shifts to ΔX, ΔY, ΔZ. The residuals are $\{v_{X,i}/S_X\}$, $\{v_{Y,i}/S_{Y,prov}\}$, $\{v_{Z,i}/S_Z\}$ which also need multiplication by the scale factors to give the true value of RMSD.

In either case (RS or SR), RMSD is minimised by treating it as a function of S_X and applying a one-dimensional optimisation method. The starting value of both S_X and S_Z is the value of S_{XZ} corresponding to $S_{Y,prov}$.

3.4. Scale-ratios correction

The final RMSD from 3.3 is the minimum for the 9-parameter affine transformation subject to the constraint $S_Y = S_{Y,prov}$ and it produced a ratio c for S_Z/S_X. The next stage removes the constraint on S_Y but requires that $S_Z = cS_X$ (*ie* treats the ratio c as final).

This enables condition (10) to be rewritten

$$S_X^2\sum d_X^2 + S_Y^2\sum d_Y^2 + \mathrm{c}^2 S_X^2\sum d_Z^2 = S_{\mathrm{DA}}^2\sum(d_X^2 + d_Y^2 + d_Z^2) \tag{20}$$

As a result, S_Y can be expressed in terms of S_X:

$$S_Y = \sqrt{\frac{S_{\mathrm{DA}}^2\sum(d_X^2+d_Y^2+d_Z^2) - S_X^2\sum(d_X^2+\mathrm{c}^2 d_Z^2)}{\sum d_Y^2}} \tag{21}$$

This substitution is applied to each trial value of S_X.

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:

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} + \mathbf{R}\begin{bmatrix} S_X X_s \\ S_Y Y_s \\ cS_X Z_s \end{bmatrix} \tag{22}$$

The input arguments to RIGOPT for a trial value of S_X would be n, $\{S_X X_{s,i}\}$, $\{S_Y Y_{s,i}\}$, $\{cS_X Z_{s,i}\}$, $\{X_{t,i}\}$, $\{Y_{t,i}\}$, $\{Z_{t,i}\}$. The first 6 output arguments are ΔX, ΔY, ΔZ, R_X, R_Y, R_Z which complete the set of affine parameters. RMSD is easily computed from the residuals.

If the affine model is of type SR, the 6-parameter rigid transformation is treated as that which transforms the source coordinates to the 'de-scaled' target coordinates:

$$\begin{bmatrix} X_t/S_X \\ Y_t/S_Y \\ Z_t/(cS_X) \end{bmatrix} = \begin{bmatrix} \Delta X/S_X \\ \Delta Y/S_Y \\ \Delta Z/(cS_X) \end{bmatrix} + \mathbf{R}\begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} \tag{23}$$

The input arguments to RIGOPT for a trial value of S_X would be $n, \{X_{s,i}\}, \{Y_{s,i}\}, \{Z_{s,i}\}, \{X_{t,i}/S_X\}, \{Y_{t,i}/S_Y\}, \{Z_{t,i}/(cS_X)\}$. The first 6 output arguments are $\Delta X/S_X$, $\Delta Y/S_Y$, $\Delta Z/(cS_X)$, R_X, R_Y, R_Z which complete the set of affine parameters after the scale factors convert the shifts to ΔX, ΔY, ΔZ. The residuals are $\{v_{X,i}/S_X\}$, $\{v_{Y,i}/S_Y\}$, $\{v_{Z,i}/(cS_X)\}$ which also need multiplication by the scale factors to give the true value of RMSD.

In either case (RS or SR), RMSD is minimised by treating it as a function of S_X and applying a one-dimensional optimisation method. The starting value of S_X is that obtained in 3.3.

3.5. Numerical verification

To verify that RMSD is minimised by the above process, it is sufficient to show that the partial derivatives of RMSD with respect to the scale factors are zero. This can be done numerically by examination of enclosing intervals. It was used in the case studies described in Section 4, and proved convergence.

Let RMSD_0 denote the RMSD obtained from applying RIGOPT to the data after multiplication of the source coordinates by S_X, S_Y and S_Z. Let RMSD_{-1} and RMSD_1 denote the RMSD obtained when S_X is replaced by $S_X - \delta S$ and $S_X + \delta S$ respectively, where δS is a suitably small perturbation of scale in the *X* direction. Then RMSD_0 can be assumed to be within tolerance *T* of the minimum with respect to scale in the *X*-direction if the following two inequalities are satisfied.

$$\mathrm{RMSD}_0 < Min(\mathrm{RMSD}_{-1}, \mathrm{RMSD}_1) \tag{24}$$

$$\frac{(\mathrm{RMSD}_1 - \mathrm{RMSD}_{-1})^2}{8(\mathrm{RMSD}_{-1} + \mathrm{RMSD}_1 - 2\mathrm{RMSD}_0)} < T \tag{25}$$

The rationale behind condition (25) is that if $Q(S)$ is the quadratic function that interpolates $(S_X - \delta S, \mathrm{RMSD}_{-1})$, (S_X, RMSD_0) and $(S_X + \delta S, \mathrm{RMSD}_1)$, then

$$Min[Q(S)] = \mathrm{RMSD}_0 - \frac{(\mathrm{RMSD}_1 - \mathrm{RMSD}_{-1})^2}{8(\mathrm{RMSD}_{-1} + \mathrm{RMSD}_1 - 2\mathrm{RMSD}_0)} \tag{26}$$

Similar tests are applied for RMSD_0 in the *Y* direction using $S_Y \pm \delta S$ and for RMSD_0 in the *Z* direction using $S_Z \pm \delta S$.

Palláncz *et al.* (2010) successfully used a generic 'Global Minimisation' (GM) algorithm to verify the results of their PZ method. The generic algorithm was the built-in function NMinimize in the software system Wolfram Mathematica. While this option should be noted, the fact remains that it was the PZ method which was recommended by Palláncz *et al.* (2010). In any case, enclosing-interval is a much simpler way of verifying a local minimum. Each test requires only two evaluations of RIGOPT, unless δS needs amending.

4. Case studies

The new method was applied to three datasets. In all three cases, the age of the data means that the ellipsoidal heights could not have come directly from Global Navigation Satellite Systems (GNSS). The positioning would have been 2+1D in nature, suggesting that more parameters than the 7 of the conformal transformation might be needed.

- Western Australia where there were 82 data points common to Australian Geodetic Datum 1984 (AGD 84) and Geocentric Datum of Australia 1994 (GDA 94), taken as source and target datums respectively. The 82 points are the 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).
- Great Britain where there were 44 data points common to Ordnance Survey Great Britain 1936 (OSGB 36) and World Geodetic System 1984 (WGS 84), taken as source and target datums respectively. Given that the data was provided by E. J. Price who received it circa 2001, the 'WGS 84' designation should be regarded as an early realisation of the datum.
- Sweden where there were 20 points common to SWEREF93 and RT90/RH70, taken as source and target datums respectively. From Jonsson *et al.* (2003, Fig. 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 geocentric Cartesian coordinates are given in Table 4.1 of Andrei (2006), but with columns in the wrong order. A correct listing is given in Table 5 of Amiri-Simkooei (2018).

4.1. Derivation of affine models

Optimisation of the RS affine transformation was performed as described in Sections 3.1 to 3.4. The British example was the only one in which the reduction in RMSD was smaller for the nine-from-eight stage (0.8565%) than for the eight-from-seven stage (4.0859%). The reductions in RMSD for the scale-ratios correction were 0.0059%, 0.0001% and 0.0017% respectively for the three case studies, which suggests that for many purposes the previous stages are sufficient. The optimal parameters and corresponding RMSD are given in Table 1.

Table 1. Parameters and quality-of-fit of the RS affine transformation models.

	W Aust	GB	Sweden
ΔX	-112.17286 m	574.21905 m	-422.59194 m
ΔY	-44.04441 m	-162.00636 m	-99.90035 m
ΔZ	144.31558 m	366.42516 m	-585.34296 m
R_X	0.08091″	-0.79647″	-0.86856″
R_Y	0.43603″	-3.07372″	-1.72456″
R_Z	0.43709″	1.57111″	7.86120″
ΔS_X	4.1676 ppm	-32.8722 ppm	1.2417 ppm
ΔS_Y	3.1003 ppm	-15.8336 ppm	1.0803 ppm
ΔS_Z	3.4700 ppm	1.7842 ppm	0.1677 ppm
RMSD	0.733450 m	2.395991 m	0.178611 m

In each case, the numerical verification method of Section 3.5 was applied to check that the partial derivatives of RMSD with respect to the scale factors were zero. Independent evidence of minimisation was available in the case of Sweden. The parameters obtained for Andrei's optimal affine transformation were -422.604 m, -99.903 m, -585.318 m, -0.868641, -1.724197, 7.861238, 1.2425 ppm, 1.0807 ppm and 0.1642 ppm. (These are taken from Column 6 of Table 4.3 in Andrei, 2006, with a change of sign for the rotations to change them from CF to PV.) They are so close to the parameters obtained above that the author computed an identical RMSD of 0.178611 metres from the residuals.

Optimisation of the SR affine transformation was performed as described in Sections 3.1 to 3.4. The British example was the only one in which the reduction in RMSD was smaller for the nine-from-eight stage (0.8561%) than for the eight-from-seven stage (4.0858%). The reductions in RMSD for the scale-ratios correction were 0.0059%, 0.0001% and 0.0017% respectively, which suggests that for many purposes the previous stages are sufficient. The optimal parameters and corresponding RMSD are given in Table 2.

Table 2. Parameters and quality-of-fit of the SR affine transformation models.

	W Aust	GB	Sweden
ΔX	-112.16861 m	574.21280 m	-422.59194 m
ΔY	-44.04511 m	-162.00678 m	-99.90035 m
ΔZ	144.31235 m	366.42688 m	-585.34296 m
R_X	0.08093″	-0.79649″	-0.86856″
R_Y	0.43612″	-3.07362″	-1.72456″
R_Z	0.43712″	1.57109″	7.86120″
ΔS_X	4.1684 ppm	-32.8719 ppm	1.2417 ppm
ΔS_Y	3.1005 ppm	-15.8340 ppm	1.0803 ppm
ΔS_Z	3.4695 ppm	1.7838 ppm	0.1677 ppm
RMSD	0.733450 m	2.396004 m	0.178611 m

In each case, the numerical verification method of Section 3.5 was applied to check that the partial derivatives of RMSD with respect to the scale factors were zero. Further evidence of minimisation can be deduced from the fact that for the Western Australian data the RMS residual distance is equivalent to a root-sum-of-squares estimate of 6.64167 m. This agrees with the best estimate computed by Paláncz *et al.* (2010) for the PZ method, which was 6.642 m. (The estimate for the ABC method was 6.788 m.)

4.2. Assessment of affine models

Although the case studies were given primarily to demonstrate the new method of derivation, the results would be incomplete without an assessment of the derived models for the given datasets. The change from a 7-parameter conformal model to a 9-parameter affine model reduces RMSD, but that does not necessarily prove that the two extra parameters are statistically significant.

To perform this analysis, two further pieces of notation are needed.

- RMS denotes the root-mean-square of the residuals between the original target coordinates and the computed coordinates. This is followed by '(7)' or '(9)' where this helps to indicate the number of parameters in the model. Since RMS is based on $3n$ differences rather than n distances, RMS = RMSD/√3.
- σ_0 denotes the standard error of an observation of unit weight, which in the absence of different weights, is $\sqrt{\mathbf{v}^{\mathrm{T}}\mathbf{v}/n_{\mathrm{DF}}}$. The product $\mathbf{v}^{\mathrm{T}}\mathbf{v}$ is the sum of the squared residuals. n_{DF} is the number of degrees of freedom which is $3n - 7$ for the conformal model and $3n - 9$ for the affine model. σ_0 is followed by '(7)' or '(9)' where this helps to indicate the number of parameters in the model.

The changes in RMS and σ_0 arising from the extra scale parameters the are shown in Tables 3 and 4.

Table 3. Difference between the conformal and RS affine transformation models.

	W Aust	GB	Sweden
RMS(7)	0.437818 m	1.454717 m	0.103668 m
RMS(9)	0.423458 m	1.383326 m	0.103121 m
Change	-3.3912%	-5.1608%	-0.5308%
$\sigma_0(7)$	0.444183 m	1.494894 m	0.110302 m
$\sigma_0(9)$	0.431423 m	1.433042 m	0.111851 m
Change	-2.9577%	-4.3161%	+1.3843%

Table 4. Difference between the conformal and SR affine transformation models.

	W Aust	GB	Sweden
RMS(7)	0.437818 m	1.454717 m	0.103668 m
RMS(9)	0.423458 m	1.383334 m	0.103121 m
Change	-3.3912%	-5.1602%	-0.5308%
$\sigma_0(7)$	0.444183 m	1.494894 m	0.110302 m
$\sigma_0(9)$	0.431423 m	1.433050 m	0.111851 m
Change	-2.9577%	-4.3155%	+1.3843%

In the case of Western Australia, the extra parameters produce a reduction of almost 3% in σ_0, which supports the advocacy of the 9-parameter method in Paláncz *et al.* (2010).

In the case of Great Britain, the extra parameters produce a reduction of 4.3% in σ_0. This demonstrates what can be achieved by the 9-parameter affine model. The parameters for Great Britain in Tables 1 and 2 are not

recommended for practical use, because there are transformations between OSGB 36 and recent realisations of WGS 84 that are based on larger and more accurate datasets (see, for example, Ordnance Survey 2018). Nevertheless, the results raise the possibility that a 9-parameter affine transformation derived from more recent data would produce a similar improvement over a conformal transformation.

In the case of Sweden, the extra parameters actually produce an increase of 1.4% in σ_0. The change from 0.110 m to 0.112 m is noted by Andrei (2006, Table 4.3). For Sweden, the 7-parameter conformal model makes more sense than the affine model. For this study, the optimal 7 parameters were found to be -419.56857 m, -99.24601 m, -591.45613 m, -0.85019″, -1.81415″, 7.85348″ and 1.0237 ppm. Andrei's interest in 8- or 9-parameter affine transformations had more to do with local level coordinates (where there was a significant reduction in σ_0) than with geocentric Cartesian coordinates.

The analysis performed above, of course, does not provide an assessment of a transformation's accuracy as a predictor. The drawback of using an entire dataset to derive a model is that there are no points available for an independent check.

One approach for estimating prediction accuracy is *cross-validation*, a process described succinctly in Berrar (2018). Given the computation involved in the new transformation-derivation method, leave-one-out cross validation is perhaps only worth considering if there are relatively few data points. Otherwise K-fold cross-validation is suggested, perhaps with K=10.

In keeping with the derivations in Spath (2004), Watson (2006), Awange *et al*. (2008) and Palancz *et al*. (2010), no attempt has been made to compute the standard deviations of the parameters. Andrei (2006) gives figures for Sweden (ibid, Table 4-3) but they seem implausibly high given that the sigma noughts for 7 parameters and 9 parameters differ by only 2%. Computation of standard deviations for non-linear models requires a special approach, such as Monte-Carlo simulations. See, for example, Wang and Luo (2021).

5. Conclusions

Unlike the previously-published methods which derive a single form of the 9-parameter affine transformation, the proposed new method covers all 8 formulations. It is applicable to both the RS and SR versions. The method can be adapted to coordinate-frame rotation convention by the substitution suggested in Section 1. The sub-algorithm of deriving a rigid transformation has two versions covering both the commonly-used order of rotations.

The new method does not rely on concepts like Groebner bases or Procrustes analysis. The formulae for distance analysis and application of the equivalent-enlargement hypothesis are simple and easy to programme.

A by-product of the proofs of optimisation (enclosing-interval analysis and comparison against Andrei, Awange *et al.*) is that the equivalent-enlargement hypothesis is vindicated. This equates the enlargement of the optimal 9-parameter affine transformation to that of the optimal conformal transformation which in turn is obtainable from distance analysis.

So far, the method has not been tested on datasets where there are large rotations of several degrees, which can occur in photogrammetry. The RIGOPT subroutine starts with a linearisation that is not designed for rotations of that magnitude. However, if an approximate estimate of rotations is known, they can be applied to the source or target coordinates depending on whether the SR or RS affine transformation is required. The derivation of parameters would then be manageable using the new method. This variation on the new method needs further research.

For any practical application, it is recommended that the standard error of an observation of unit weight for the affine transformation is compared with that of the 7-parameter conformal transformation. If there is a reduction, then there is a statistical reason for supposing that the extra scaling parameters are significant, and that the 9-parameter model will predict transformed coordinates more accurately.

Acknowledgements

The author thanks Professor Joseph Awange of Curtin University for providing control data for Western Australia, and Ordnance Survey for providing control data for Great Britain. The author is grateful to the anonymous reviewers for their comments and suggestions.

Disclosure statement

No potential conflict of interest was reported by the author.

ORCID

A. C. Ruffhead http://orcid.org/0000-0002-7836-6702

Notes on contributor

Andrew Ruffhead is a Visiting Fellow in the School of Architecture, Computing and Engineering, University of East London, where he completed his PhD researching geodetic datum transformations. He was previously a Senior Scientific Officer at the Defence Geographic Centre, U.K., working for 25 years in computing and geodesy.

References

Agustan, A., 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. Available from: https://www.fig.net/resources/proceedings/fig_proceedings/jakarta/papers/ts_17/ts_17_3_agustan_featherstone.pdf

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.

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.

Berrar, D., 2018. Cross-validation. *Encyclopedia of Bioinformatics and Computational Biology, Volume 1*, Elsevier, 542–545.

Fan, H., 2009. Direct solution of three-dimensional coordinate transformation parameters. Department of Geodesy, Royal Institute of Technology (KTH), Sweden.

Han, J.-Y., 2010. Noniterative approach for solving the indirect problems of linear reference frame transformations. *Journal of Surveying Engineering*, 136 (4), 150-156.

ISO, 2007. *ISO 19111, Geographical information – Spatial referencing by coordinates*, Second Edition, International Organization for Standardization, Geneva, Switzerland.

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.

Ordnance Survey, 2018. *A guide to coordinate systems in Great Britain*. Version 3.3. Available from: https://www.ordnancesurvey.co.uk/docs/support/guide-coordinate-systems-great-britain.pdf?awc=2495_1472758581_2be7907c343c32b09a8d5171103197d7

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. Available from: 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.

Ruffhead, A. C., 2021. Derivation of rigorously-conformal 7-parameter 3D geodetic datum transformations. *Survey Review*, 53 (376) 35-43.

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.

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.

Wang, L., and Luo, X., 2021. Adaptive Quasi-Monte Carlo method for nonlinear function error propagation and its application in geodetic measurement. *Measurement*, 186, Article 110122, 10 pages.

Watson, G. A., 2006. Computing Helmert transformations. *Journal of Computational and Applied Mathematics*, 197 (2), 387-394.

Appendix A: Derivation of rigid transformation

The algorithm here optimises the rigid transformation $(x_s, y_s, z_s) \rightarrow (x_t, y_t, z_t)$ from coordinates of n points in source (s) and target (t) coordinate systems. It is a two-stage process based on the steps HO2 and HO3 described in Ruffhead (2021) except for the absence of scaling between the coordinate systems.

The subroutine RIGOPT has input arguments $n, \{x_{s,i}\}, \{y_{s,i}\}, \{z_{s,i}\}, \{x_{t,i}\}, \{y_{t,i}\}, \{z_{t,i}\}$. In the arrays, *i* spans the range 1 to *n*.

The output arguments consist of the parameters Δx, Δy, Δz, R_x, R_y, R_z (where the last three are in radians) and the residuals $\{v_{x,i}\}, \{v_{y,i}\}, \{v_{z,i}\}$. In the arrays, *i* spans the range 1 to *n*.

For the sake of brevity, $\cos R_x$ is denoted by c_x, $\sin R_x$ by s_x, etc.

The form of the rigid transformation is

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} + \mathbf{R} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A1}$$

In the style of equation (1), **R** is given by

$$\mathbf{R} = \begin{bmatrix} c_z & -s_z & 0 \\ s_z & c_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_y & 0 & s_y \\ 0 & 1 & 0 \\ -s_y & 0 & c_y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_x & -s_x \\ 0 & s_x & c_x \end{bmatrix} \tag{A2}$$

In the style of equation (3), **R** can be rewritten as follows:

$$\mathbf{R} = \begin{bmatrix} c_y c_z & s_x s_y c_z - c_x s_z & s_x s_z + c_x s_y c_z \\ c_y s_z & c_x c_z + s_x s_y s_z & c_x s_y s_z - s_x c_z \\ -s_y & s_x c_y & c_x c_y \end{bmatrix} \tag{A3}$$

To enable reference to be made to individual elements of **R**, the identities (A1) and (A3) can also be expressed as follows.

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} + \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A4}$$

$$\begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix} = \begin{bmatrix} c_y c_z & s_x s_y c_z - c_x s_z & s_x s_z + c_x s_y c_z \\ c_y s_z & c_x c_z + s_x s_y s_z & c_x s_y s_z - s_x c_z \\ -s_y & s_x c_y & c_x c_y \end{bmatrix} \tag{A5}$$

The initial stage is to compute starting approximations to Δx, Δy, Δz, R_x, R_y, R_z, which will be denoted by $\overline{\Delta x}$, $\overline{\Delta y}$, $\overline{\Delta z}$, $\bar{R}_x$, $\bar{R}_y$, $\bar{R}_z$. Using $\bar{r}_{i,j}$ to denote the corresponding approximations in **R**, the identity (A5) becomes

$$\begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} = \begin{bmatrix} 1 & -\bar{R}_z & \bar{R}_y \\ \bar{R}_z & 1 & -\bar{R}_x \\ -\bar{R}_y & \bar{R}_x & 1 \end{bmatrix} \tag{A6}$$

The linearised version of (A4) becomes

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = \begin{bmatrix} \overline{\Delta x} \\ \overline{\Delta y} \\ \overline{\Delta z} \end{bmatrix} + \begin{bmatrix} 1 & -\bar{R}_z & \bar{R}_y \\ \bar{R}_z & 1 & -\bar{R}_x \\ -\bar{R}_y & \bar{R}_x & 1 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A7}$$

This leads to the following observation equations, in which the final column vector consists of residuals.

$$\begin{bmatrix} x_{t,i} - x_{s,i} \\ y_{t,i} - y_{s,i} \\ z_{t,i} - z_{s,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & z_{s,i} & -y_{s,i} \\ 0 & 1 & 0 & -z_{s,i} & 0 & x_{s,i} \\ 0 & 0 & 1 & y_{s,i} & -x_{s,i} & 0 \end{bmatrix} \begin{bmatrix} \overline{\Delta x} \\ \overline{\Delta y} \\ \overline{\Delta z} \\ \bar{R}_x \\ \bar{R}_y \\ \bar{R}_z \end{bmatrix} + \begin{bmatrix} v_{x,i} \\ v_{y,i} \\ v_{z,i} \end{bmatrix} \text{ for } i = 1,\ldots,n \tag{A8}$$

These equations are linear with respect to $\overline{\Delta x}$, $\overline{\Delta y}$, $\overline{\Delta z}$, $\bar{R}_x$, $\bar{R}_y$, $\bar{R}_z$, and it is a straightforward exercise to compute their values by ordinary least-squares. The last three quantities are fed into the corrective stage, described below.

The approximate rotations give rise to approximate values $\bar{r}_{i,j}$ of the elements $r_{i,j}$ defined in (A5). Using $\bar{c}_x$ to denote $\cos\bar{R}_x$ and $\bar{s}_x$ to denote $\sin\bar{R}_x$, etc,

$$\begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} = \begin{bmatrix} \bar{c}_Y \bar{c}_Z & \bar{s}_X \bar{s}_Y \bar{c}_Z - \bar{c}_X \bar{s}_Z & \bar{s}_X \bar{s}_Z + \bar{c}_X \bar{s}_Y \bar{c}_Z \\ \bar{c}_Y \bar{s}_Z & \bar{c}_X \bar{c}_Z + \bar{s}_X \bar{s}_Y \bar{s}_Z & \bar{c}_X \bar{s}_Y \bar{s}_Z - \bar{s}_X \bar{c}_Z \\ -\bar{s}_Y & \bar{s}_X \bar{c}_Y & \bar{c}_X \bar{c}_Y \end{bmatrix} \tag{A9}$$

The corrections δR_x, δR_y, δR_z that need to be added to the approximate rotations are defined as follows.

$$\begin{bmatrix} R_x \\ R_y \\ R_z \end{bmatrix} = \begin{bmatrix} \bar{R}_x \\ \bar{R}_y \\ \bar{R}_z \end{bmatrix} + \begin{bmatrix} \delta R_x \\ \delta R_y \\ \delta R_z \end{bmatrix}. \tag{A10}$$

Then each term in the rotation matrix **R** can be expressed in the form

$$r_{i,j} = \bar{r}_{i,j} + \delta R_x(\partial r_{i,j}/\partial R_x) + \delta R_y(\partial r_{i,j}/\partial R_y) + \delta R_z(\partial r_{i,j}/\partial R_z) \tag{A11}$$

In keeping with the standard iterative processes of numerical analysis, the partial derivatives in (A11) are evaluated at $\bar{R}_X$, $\bar{R}_Y$, $\bar{R}_Z$, and higher-order terms involving δR_x, δR_y, δR_z are ignored.

This enables (A4) to be rewritten in the form

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} + \begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} + \begin{bmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,1} & m_{3,2} & m_{3,3} \end{bmatrix} \begin{bmatrix} \delta R_x \\ \delta R_y \\ \delta R_z \end{bmatrix} \tag{A12}$$

Applying (A11), the terms $m_{i,j}$ can be evaluated from the following equations.

$$m_{i,1} = (\partial r_{i,1}/\partial R_x)x_s + (\partial r_{i,2}/\partial R_x)y_s + (\partial r_{i,3}/\partial R_x)z_s \tag{A13}$$

$$m_{i,2} = (\partial r_{i,1}/\partial R_y)x_s + (\partial r_{i,2}/\partial R_y)y_s + (\partial r_{i,3}/\partial R_y)z_s \tag{A14}$$

$$m_{i,3} = (\partial r_{i,1}/\partial R_z)x_s + (\partial r_{i,2}/\partial R_z)y_s + (\partial r_{i,3}/\partial R_z)z_s \tag{A15}$$

From (A13), the partial derivatives with respect to R_x give the following:

$$\begin{bmatrix} m_{1,1} \\ m_{2,1} \\ m_{3,1} \end{bmatrix} = \begin{bmatrix} 0 & \bar{c}_x \bar{s}_y \bar{c}_z + \bar{s}_x \bar{s}_z & \bar{c}_x \bar{s}_z - \bar{s}_x \bar{s}_y \bar{c}_z \\ 0 & \bar{c}_x \bar{s}_y \bar{s}_z - \bar{s}_x \bar{c}_z & -(\bar{s}_x \bar{s}_y \bar{s}_z + \bar{c}_x \bar{c}_z) \\ 0 & \bar{c}_x \bar{c}_y & -\bar{s}_x \bar{c}_y \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A16}$$

From (A14), the partial derivatives with respect to R_y give the following:

$$\begin{bmatrix} m_{1,2} \\ m_{2,2} \\ m_{3,2} \end{bmatrix} = \begin{bmatrix} -\bar{s}_y \bar{c}_z & \bar{s}_x \bar{c}_y \bar{c}_z & \bar{c}_x \bar{c}_y \bar{c}_z \\ -\bar{s}_y \bar{s}_z & \bar{s}_x \bar{c}_y \bar{s}_z & \bar{c}_x \bar{c}_y \bar{s}_z \\ -\bar{c}_y & -\bar{s}_x \bar{s}_y & -\bar{c}_x \bar{s}_y \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A17}$$

From (A15), the partial derivatives with respect to R_z give the following:

$$\begin{bmatrix} m_{1,3} \\ m_{2,3} \\ m_{3,3} \end{bmatrix} = \begin{bmatrix} -\bar{c}_y\bar{s}_z & -(\bar{s}_x\bar{s}_y\bar{s}_z + \bar{c}_x\bar{c}_z) & \bar{s}_x\bar{c}_z - \bar{c}_x\bar{s}_y\bar{s}_z \\ \bar{c}_y\bar{c}_z & \bar{s}_x\bar{s}_y\bar{c}_z - \bar{c}_x\bar{s}_z & \bar{c}_x\bar{s}_y\bar{c}_z + \bar{s}_x\bar{s}_z \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A18}$$

That only leaves Δx, Δy, Δz, δR_x, δR_y and δR_z as the unknowns in (A12). This leads to the following observation equations, in which the final column vector consists of residuals.

$$\begin{bmatrix} x_{t,i} \\ y_{t,i} \\ y_{t,i} \end{bmatrix} - \begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} \begin{bmatrix} x_{s,i} \\ y_{s,i} \\ z_{s,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & m_{1,1,i} & m_{1,2,i} & m_{1,3,i} \\ 0 & 1 & 0 & m_{2,1,i} & m_{2,2,i} & m_{2,3,i} \\ 0 & 0 & 1 & m_{3,1,i} & m_{3,2,i} & m_{3,3,i} \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \delta R_x \\ \delta R_y \\ \delta R_z \end{bmatrix} + \begin{bmatrix} v_{x,i} \\ v_{y,i} \\ v_{z,i} \end{bmatrix} \text{ for } i = 1,\ldots,n \tag{A19}$$

These equations are linear with respect to Δx, Δy, Δz, δR_x, δR_y, δR_z, and it is a straightforward exercise to compute their values by ordinary least-squares. Application of (A10) completes the computation of R_x, R_y, R_z. The corrective stage would only need repeating if any of the corrections δR_x, δR_y, δR_z were substantial.

For completeness, the following variations are noted in the case where the rotation matrix is based on equations (2) and (4). That is to say

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_x & -s_x \\ 0 & s_x & c_x \end{bmatrix} \begin{bmatrix} c_y & 0 & s_y \\ 0 & 1 & 0 \\ -s_y & 0 & c_y \end{bmatrix} \begin{bmatrix} c_z & -s_z & 0 \\ s_z & c_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A20}$$

In the style of equation (3), $\mathbf{R}$ can be rewritten as follows:

$$\mathbf{R} = \begin{bmatrix} c_y c_z & c_y s_z & s_y \\ s_x s_y c_z + c_x s_z & c_x c_z - s_x s_y s_z & -s_x c_y \\ s_x s_z - c_x s_y c_z & c_x s_y s_z + s_x c_z & c_x c_y \end{bmatrix} \tag{A21}$$

The initial stage would be unaffected. For the corrective stage, only equations (A9), (A13), (A14) and (A15) would need to be replaced:

$$\begin{bmatrix} \bar{r}_{1,1} & \bar{r}_{1,2} & \bar{r}_{1,3} \\ \bar{r}_{2,1} & \bar{r}_{2,2} & \bar{r}_{2,3} \\ \bar{r}_{3,1} & \bar{r}_{3,2} & \bar{r}_{3,3} \end{bmatrix} = \begin{bmatrix} \bar{c}_y\bar{c}_z & -\bar{c}_y\bar{s}_z & \bar{s}_y \\ \bar{s}_x\bar{s}_y\bar{c}_z + \bar{c}_x\bar{s}_z & \bar{c}_x\bar{c}_z - \bar{s}_x\bar{s}_y\bar{s}_z & -\bar{s}_x\bar{c}_y \\ \bar{s}_x\bar{s}_z - \bar{c}_x\bar{s}_y\bar{c}_z & \bar{c}_x\bar{s}_y\bar{s}_z + \bar{s}_x\bar{c}_z & \bar{c}_x\bar{c}_y \end{bmatrix} \tag{A22}$$

$$\begin{bmatrix} m_{1,1} \\ m_{2,1} \\ m_{3,1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ \bar{c}_x\bar{s}_y\bar{c}_z - \bar{s}_x\bar{s}_z & -(\bar{s}_x\bar{c}_z + \bar{c}_x\bar{s}_y\bar{s}_z) & -\bar{c}_x\bar{c}_y \\ \bar{c}_x\bar{s}_z + \bar{s}_x\bar{s}_y\bar{c}_z & \bar{c}_x\bar{c}_z - \bar{s}_x\bar{s}_y\bar{s}_z & -\bar{s}_x\bar{c}_y \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A23}$$

$$\begin{bmatrix} m_{1,2} \\ m_{2,2} \\ m_{3,2} \end{bmatrix} = \begin{bmatrix} -\bar{s}_y\bar{c}_z & \bar{s}_y\bar{s}_z & \bar{c}_y \\ \bar{s}_x\bar{c}_y\bar{c}_z & -\bar{s}_x\bar{c}_y\bar{s}_z & \bar{s}_x\bar{s}_y \\ -\bar{c}_x\bar{c}_y\bar{c}_z & \bar{c}_x\bar{c}_y\bar{s}_z & -\bar{c}_x\bar{s}_y \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A24}$$

$$\begin{bmatrix} m_{1,3} \\ m_{2,3} \\ m_{3,3} \end{bmatrix} = \begin{bmatrix} -\bar{c}_y\bar{s}_z & -\bar{c}_y\bar{c}_z & 0 \\ \bar{c}_x\bar{c}_z - \bar{s}_x\bar{s}_y\bar{s}_z & -(\bar{c}_x\bar{s}_z + \bar{s}_x\bar{s}_y\bar{c}_z) & 0 \\ \bar{s}_x\bar{c}_z + \bar{c}_x\bar{s}_y\bar{s}_z & \bar{c}_x\bar{s}_y\bar{c}_z - \bar{s}_x\bar{s}_z & 0 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{A25}$$