An ANFIS-based cache replacement method for mitigating cache pollution attacks in Named Data Networking

Amin Karami*, Manel Guerrero-Zapata

Computer Architecture Department (DAC), Universitat Politècnica de Catalunya (UPC), Campus Nord, Jordi Girona 1-3, 08034 Barcelona, Spain

Abstract

Named Data Networking (NDN) is a candidate next-generation Internet architecture designed to overcome the fundamental limitations of the current IP-based Internet, in particular strong security. The ubiquitous in-network caching is a key NDN feature. However, pervasive caching strengthens security problems namely cache pollution attacks including cache poisoning (i.e., introducing malicious content into caches as false-locality) and cache pollution (i.e., ruining the cache locality with new unpopular content as locality-disruption).
In this paper, a new cache replacement method based on Adaptive Neuro-Fuzzy Inference System (ANFIS) is presented to mitigate the cache pollution attacks in NDN. The ANFIS structure is built using the input data related to the inherent characteristics of the cached content and the output related to the content type (i.e., healthy, locality-disruption, and false-locality). The proposed method detects both false-locality and locality-disruption attacks as well as a combination of the two on different topologies with high accuracy, and mitigates them efficiently without very much computational cost as compared to the most common policies.

Keywords: Named Data Networking, False-locality, Locality-disruption, Cache replacement, ANFIS

1. Introduction

Content-Centric Networking (CCN) has recently been considered as a promising architecture for the next-generation Internet, shifting from sender-driven end-to-end communication paradigm to receiver-driven content retrieval paradigm [1, 2, 3]. This change from host-centric to content-centric has several significant advantages such as network load reduction, low dissemination latency, scalability, etc. Additionally, strong security has been one of the main design requirements for these architectures [4, 5, 6].
Named Data Networking (NDN) [7] is a prominent example and ongoing research effort of CCN design. In NDN, a consumer asks for a *Content* by sending an *Interest* request towards the location of the content's origin where it has been published using name prefix (content identifier) instead of today's IP prefix (content location). The network can satisfy an Interest packet with a requested content from any intermediate node that holds a copy of the content in its cache (*Content Store*). On the way back, -in reverse, the exact path of the proceeding Interest- all the intermediate nodes store a copy of the content in their caches to satisfy subsequent users interested in that content (i.e., in-network caching) [8, 9, 10]. The ubiquitous in-network caching is a key NDN feature as it reduces overall latency and improves bandwidth utilization for popular content [11, 12, 13, 14, 15]. However, pervasive caching strengthens the security problem of cache pollution attacks in two generic classes: locality-disruption and false-locality [16, 17, 18, 19]. Locality-disruption attacks continuously generate requests for new unpopular files to force routers (i.e., the victims of the attack) to cache unpopular content, thus degrading cache efficiency by ruining the cache file locality. False-locality attacks repeatedly request the same set of unpopular (i.e., fake popular) files, thus degrading the hit ratio by creating a false file locality at cache.
Cache replacement algorithms play an important role in the analysis of cache pollution attacks [20, 21, 22]. Cache replacement refers to the process that a cache capacity becomes full and old content must be removed to

*Corresponding author, Telephone: 0034-934011638
Email addresses: amin@ac.upc.edu (Amin Karami), guerrero@ac.upc.edu (Manel Guerrero-Zapata)
URL: http://personals.ac.upc.edu/amin (Amin Karami), http://personals.ac.upc.edu/guerrero (Manel Guerrero-Zapata)

make a space for new content. However, the most replacement algorithms and policies are susceptible to a subclass of pollution attacks [21, 22]. These algorithms and policies consider just one criterion and ignore other criteria that may influence on the caching efficiency and suffer from cache pollution attacks [23, 24, 25]. In this paper, a new cache replacement method in NDN is developed to detect and mitigate these two types of cache pollution attacks. The proposed method is based on the relationship between inherent characteristics of the cached content and the content type (i.e., attack or non-attack). Many researchers have proposed meaningful relationship between a series of nonlinear input-output data patterns using Adaptive Neuro-Fuzzy Inference System (ANFIS) [26, 27, 28, 29]. ANFIS is a beneficial method to handle linguistic concepts and find nonlinear relationships between inputs and outputs, which is a combination of the strength of Artificial Neural Network (ANN) and fuzzy systems [30, 31]. In ANFIS, neural networks extract automatically fuzzy rules from numerical data through the learning process, and the membership functions are adaptively adjusted. The whole proposed ANFIS-based cache replacement method contains three steps: the input-output data patterns are extracted from the NDN scenarios at first. The input features are the inherent characteristics and statistical data of the cached content, and the output is the numerical value which refer to the type of the content, i.e., locality-disruption, false-locality or healthy. After that, the accuracy of constructed ANFIS is verified under different cache pollution circumstances. And finally, the constructed model is established in a simulation environment to be integrated with NDN topologies as a novel cache replacement method to mitigate cache pollution attacks in a timely manner.

The main objective of the proposed method is to enable the caching efficiency through a novel nonlinear cache replacement method in the presence of the cache pollution attacks and satisfy some applied performance metrics. The evaluation through simulations shows that the proposed nonlinear cache replacement method based on ANFIS provides benefits in cache robustness and mitigating cache pollution attacks with high accuracy in a timely manner. We then illustrate that the proposed method provides a suitable compromise between overhead and applied performance metrics as compared to some common existing countermeasures.

The reminder of this paper is constructed as follows. In Section 2, a short overview of NDN is presented. Section 3 presents previous works on cache pollution attacks and countermeasures in NDN. ANFIS architecture is outlined in Section 4. The proposed ANFIS-based cache replacement method for mitigating cache pollution attacks is introduced in Section 5. Experimental setup, observations and results of the proposed method are studied in detail in Sections 6 and 7, respectively. At last, the conclusion is given in Section 8.

2. NDN overview

All communication in NDN is performed using two types of packets: *Interest* and *Content (Data)*. Both types of packets carry a name, which uniquely identifies a piece of data that can be carried in one data packet [7]. Data names in NDN are hierarchically structured, e.g., seven fragment of a YouTube video would be named `/youtube/videos/B87Rdx9s2/7`. Each NDN node maintains three major data structures: *Content Store*, *Pending Interest Table (PIT)*, and *Forwarding Information Base (FIB)*. To retrieve a content, a consumer sends an Interest packet using name prefix instead of today's IP prefix towards the location of the content's origin where it has been published. When a NDN router receives an Interest packet, it first checks its content store (i.e., temporary cache of Data packets for data retrieval efficiency). If there is no the requested content, the PIT table records the Interest's name, incoming interface(s), and outgoing interface(s) and then forwards the Interest packet using its FIB towards potential data sources. The PIT table holds all not yet satisfied Interest packets. The FIB is a name-prefix-based routing table to determine interfaces for forwarding incoming Interest packets based on the forwarding strategies. When an Interest packet is satisfied, on the reverse path of Interest, all the intermediate nodes remove the corresponding PIT table entry and cache the content in their content store to answer to probable same Interest requests from subsequent consumers [4, 32].

3. Related work

As a new Internet architecture proposal, there is very limited work recently regarding to mitigation of cache pollution attacks in NDN. Park et al. [18] propose a detection approach against locality disruption attacks using randomness checks of a matrix in CCN. They apply a filtering approach and a statistical sequential analysis (i.e., cumulative sum (CUSUM) algorithm) to detect low-rate attacks. Since the analysis is based on a very simple CCN scenario, the results cannot be extented to a larger CCN topology. Conti et al. [20] introduce a lightweight detection technique for detecting locality-disruption attacks. However, authors do not apply any reaction method for mitigating attacks. Xie et

al. [17] introduce a technique, called CacheShield with the goal of improving NDN cache robustness against locality disruption attacks. In CacheShield, when a router receives a content object, the CS evaluates a shielding function based on a logistic function that determines whether the content object should be cached. The CacheShield must run continuously even when no attack is in progress and store a large amount of statistics at each router that may reduce the space available to cache content. Paper [20] shows that CacheShield is ineffective against some pollution attacks and introduces new attacks specific to CacheShield. Ghali et al. [19] propose a ranking algorithm for cached content that allows routers to probabilistically distinguish good and bad content. This ranking is based on statistics collected from consumers' actions following delivery of content objects. Authors evaluate the performance of their ranking algorithm with inactive adversaries. They also assume that any fake content has a valid version till the proposed algorithm detects fake versions. The ranking algorithm must store several versions of the same content to detect a valid version, and therefore consume routers' storage and computing resources such as FIB for returning back the different possible versions of a same content.

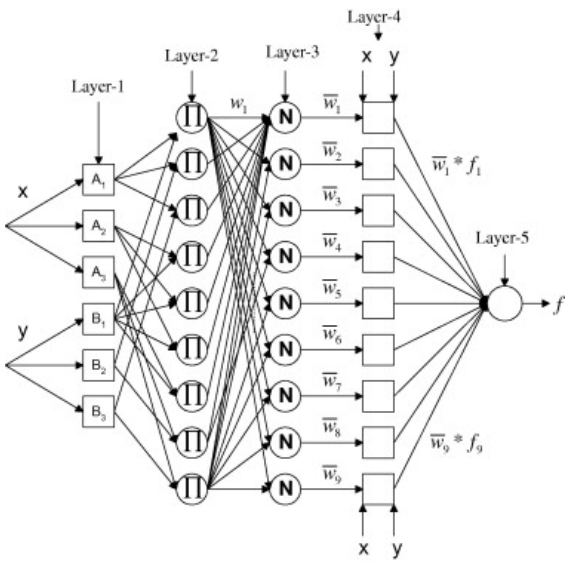


Figure 1: ANFIS architecture with two inputs and nine rules

4. ANFIS

ANFIS is a class of adaptive networks whose functionality is equivalent to a fuzzy inference system which generates a fuzzy rule base and membership functions automatically [33]. ANFIS is an integration of neural network architectures with fuzzy inference system (FIS) to map a couple of inputs-output data patterns. An ANFIS constructs a FIS (*if-then* rules) whose membership function parameters are adjusted using either backpropagation algorithm or in combination with a least squares type of method [34, 35]. An ANFIS architecture consists of a fuzzy layer, product layer, normalized layer, defuzzy layer, and summation layer. A typical architecture of ANFIS with two inputs (x and y), nine rules and one output (f) is depicted in Fig. 1[1]. Among many FIS models, the 1st order Sugeno fuzzy model is the most widely applied adaptive technique with high interpretability and computational efficiency for different problems [28, 34]. For a 1st order of Sugeno fuzzy model, a typical rule set with two fuzzy *if-then* rules can be expressed as:

$$if\, x\ is\ A_1\ and\ y\ is\ B_1\ then\ f_1 = p_1 x + q_1 y + r_1 \quad (1)$$

$$if\, x\ is\ A_2\ and\ y\ is\ B_2\ then\ f_2 = p_2 x + q_2 y + r_2 \quad (2)$$

where A_i and B_i are the fuzzy sets in the antecedent and p_i, q_i, and r_i are the linear output parameters that are determined during the training process. As in Fig. 1, an ANFIS consists of five layers and nine *if-then* rules as follows:

Layer-1: all the square nodes in this layer are adaptive nodes. The outputs of layer 1 are the fuzzy membership grade of the inputs, which are given by:

$$O_{1,i} = \mu_{Ai}(x),\ for\ i = 1,2,3\ \ O_{1,i} = \mu_{Bi-3}(y),\ for\ i = 4,5,6 \quad (3)$$

where x and y are inputs to node i, and A_i and B_i are linguistic labels for inputs. $O_{1,i}$ is the membership function of A_i and B_i. $\mu_{Ai}(x)$ and $\mu_{Bi-3}(y)$ can adopt any fuzzy membership function. For instance, if a Gaussian membership function is employed:

$$\mu_{Ai}(x)\ ,\ \mu_{Bi-3}(y) = exp\left[-(\frac{x - c_i}{a_i})^2\right] \quad (4)$$

where c_i and a_i are the parameter set of the membership function. These parameters in this layer are referred to a premise parameters.

Layer-2: Every node in this layer is a fixed node with a

[1]Reprinted from Expert Systems with Applications, Vol 37/12, Melek Acar Boyacioglu and Derya Avci, An Adaptive Network-Based Fuzzy Inference System (ANFIS) for the prediction of stock market return: The case of the Istanbul Stock Exchange, pages 7908-7912, Copyright (2010), with permission from Elsevier

circle node label $\prod$ which multiplies the incoming signals and sends the product out. The output of this layer can be represented as:

$$O_{2,i} = w_i = \mu_{Ai}(x) \times \mu_{Bi-3}(y), \quad i = 1, 2, 3, ..., 9 \qquad (5)$$

Each node output represents the firing strength of a rule.
Layer-3: Every node in this layer is also a fixed circle node labeled N, indicating that they play a normalization role to the firing strengths from the previous layer. The outputs of this layer can be represented as:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{(w_1 + w_2 + ... + w_9)}, \quad i = 1, 2, 3, ..., 9 \quad (6)$$

Layer-4: In this layer, the square nodes are adaptive nodes. The output of each node in this layer is the product of the normalized firing strength and a 1st order polynomial (for a 1st order Sugeno model):

$$O_{4,i} = \bar{w}_i \, . \, f_i = w_i \, . \, (p_i x + q_i y + r_i), \quad i = 1, 2, 3, ..., 9 \quad (7)$$

where w_i is the output of layer 3 and three parameters $\{p_i,\ q_i,\ r_i\}$ are the parameter set which will be referred to as consequent parameters.
Layer-5: The single node in this layer is a circle node labeled $\sum$ (overall output) that performs the summation of all incoming signals:

$$O_{5,i} = f = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \qquad (8)$$

ANFIS has a hybrid learning rule algorithm which integrates the gradient descent and the least squares methods to train and adjust the premise and consequent parameters [36]. The hybrid learning algorithm is composed of a forward pass and a backward pass. The least squares method (forward pass) is used to optimize the consequent parameters with the premise parameters fixed to minimize the measured error in layer 4. In the backward pass, the premise parameters are updated by the gradient descent method [33, 36].

5. An ANFIS-based cache replacement method for mitigating cache pollution attacks

In this section, we design and construct the material of proposed ANFIS-based cache replacement method for mitigating cache pollution attacks in NDN. Afterwards, we use simulation to evaluate the effectiveness of the proposed method in two considered NDN topologies in Fig. 3. The proposed ANFIS-based cache replacement architecture is depicted in Fig. 2. The detail of the method is proposed as follows.

5.1. Data preparation

The detection of cache pollution attacks is hard because all requested content are uncorrupted. The traditional detection methods usually observe and learn the legitimate users' traffic patterns and detect attacks and anomalies when such patterns change. To address this challenge, we analyze the inherent characteristics of cache pollution attacks and design a nonlinear approximation function through ANFIS to detect locality-disruption and false-locality attacks separately. In order to formulate the problem and construct an ideal method regarding the relationship between inputs (i.e., inherent characteristics of cached content) and output (i.e., content type) data, we define a set of parameters that govern the proposed ANFIS-based cache replacement method. We extract the input parameters based on published research articles such as [19, 37, 38], our observation during the design, and experts' opinion. The considered input parameters are defined as follows:

1. The cached content's longevity (*Longevity*). This corresponds to the time that content has remained in the cache between the time of content being cached and the current time.
2. The cached content's frequency access (*Frequency*). This corresponds to the estimation of content's access frequency. An Exponentially Weighted Moving Average (EWMA) method is employed as a filter to obtain a recent estimate of the access frequency rate. It can also identify the possible aberrant behavior of content's access frequency. EWMA applies weighting factors which decrease exponentially. The weighting for each older data decreases exponentially, giving much more importance to recent observations while still not discarding older observations entirely [39]. The degree of weighting decrease is expressed as a constant smoothing factor β, a number between 0 and 1. EWMA formula is defined as:

$$\overline{\mu}_n = \beta \overline{\mu}_{n-1} + (1 - \beta) x_n \qquad (9)$$

   where $\overline{\mu}_n$ is the exponentially weighted moving average of the past measurements and x_n is the number of content's access frequency in the n-th time interval. We apply the six recent time intervals (i.e., each 0.25 second) to calculate EWMA efficiently.
3. The Standard Deviation of content's access frequency in recent six time intervals (*Std.*). This parameter allows to distinguish the type of content

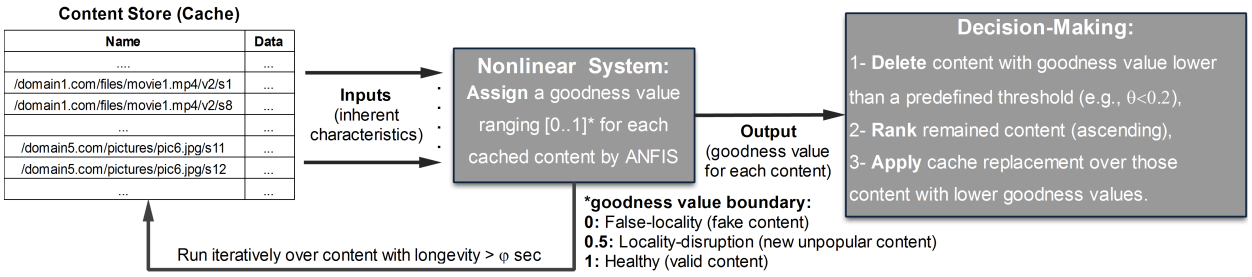
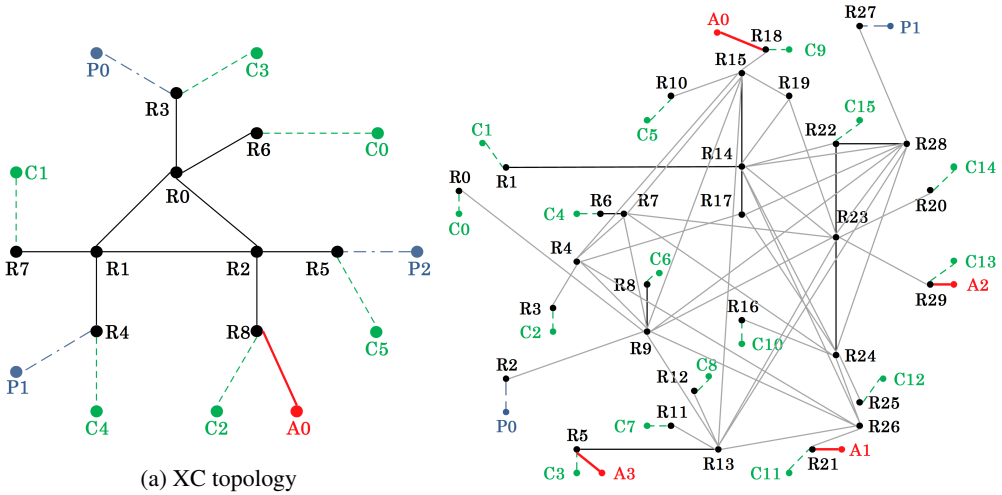


Figure 2: Schematic of the proposed ANFIS-based cache replacement method in NDN



(a) XC topology

(b) DFN topology

Figure 3: Considered network topologies

request distribution. The Std. of an uniform distribution is (close to) zero, while other types of distribution such as normal and skewed are not (close to) zero.

4. The last access to the content (*Last Retrieval*). It corresponds to the time interval between the last time of content being used and the current time.
5. The percentage of cache hit (*Hit Ratio*). It corresponds to the content cache hit vs. the total content cache hit in the recent time interval.
6. The variance of an entire population of repeated requests for a same content from the local interfaces (*Interface turnout*). The variance allows to detect distributed cache pollution attacks, when all local interfaces return the same content continuously. If all local interfaces return the similar rate of content, the variance is close to zero.

The output of each data pattern is a goodness value which determines the type of content ranging [0..1]. The boundary of assigned goodness value is defined as: 0 (false-locality or fake content), 0.5 (locality-disruption or new unpopular content), and 1 (healthy). We apply EWMA criterion in the last three time intervals to calculate the average goodness value for each content over a period of time.

According to Fig. 2, the applied ANFIS model is automatically executed after every time interval (we set it to 1 second) over cached content with longevity more than a threshold (we set 0.25 second) in order to rank all content based on the goodness value. The initial goodness value for new incoming content with the longevity less than the threshold is set to one (healthy content). After running ANFIS, each content gets a goodness value from the healthy (good) to fake (bad). Those content with the goodness value less than a predefined threshold (we set $\theta < 0.2$) are removed from the cache due to their fake type. This allows to possible valid (healthy) content be replaced with the fake version. Then, the re-

maining content is sorted in ascending order based on the goodness value for cache replacement when a new content enters and the cache has not enough space for storing. Thus, the proposed method can efficiently and accurately mitigate the false-locality (by removing the fake content) and the locality-disruption (by removing those content with the lower goodness value for cache replacement) attacks in a timely manner.

5.2. Materials of ANFIS

During the training process, ANFIS tries to minimize the training error between the target output (i.e., the type of cached content) and the actual output of the ANFIS. The input-output data samples are collected based on the section 5.1. In particular, we set cache size to infinite to not to apply any cache replacement algorithm during the training process (see section 6).
In this work, ANFIS is established in MATLAB environment. The ANFIS has 6 inputs and one output. Before the training process, data samples should be normalized into [0..1], when dealing with parameters of different units and scales [40, 41]. All variables of ANFIS have "gaussmf" membership function. The gaussmf is a kind of smooth membership functions, so the resulting model has a high accuracy [42]. The "and", "or" and "defuzzification" methods in ANFIS are selected as "product", "max" and "center of gravity", respectively. Fig. 4 shows the structure of the constructed ANFIS model as well as the number of fuzzy if-then rules. Before training process, the ANFIS structure is initialized by the fuzzy c-mean method through the input-output data, and its parameters are optimized by least squares and gradient descent algorithms.

6. Experimental setup

This section describes the considered network topologies, simulation environment, followed by the modeling of cache pollution attack strategies.

6.1. Simulation environment

We evaluate cache pollution attacks and countermeasures discussed in this paper via simulations. We rely on open-source ndnSIM [43] package, a module for ns-3 developed at UCLA as part of the NDN project. The ANFIS-based cache replacement method was firstly implemented with MATLAB on an Intel Pentium 4 3.0 GHz CPU, 4 GB RAM running Windows 7 Ultimate. Then, it was compiled as a C++ shared library using the MATLAB compiler in order to integrated it with the ndnSIM environment.

The Experiments are performed over two topologies, as illustrated in Fig. 3[2]: Xie-complex (XC) and the German Research Network (DFN). The XC and DFN topologies have been identified in previous works as meaningful topologies for simulation [20, 44]. There are several commonly used symbols to identify the type of nodes in NDN networks (such as Fig. 3), including `Cx`, `Px`, `Rx`, and `Ax` to represent `x`-th consumer, producer, router and adversary nodes, respectively [20, 32]. In our configurations, we set nodes' PIT size to [500..800] entries randomly. The Interest expiration time was also set to the default timeout of 4000 ms. We set the link delay and queue length parameters to fixed values for every node. In particular, we set delay and queue length to 10 ms and 500, respectively. The requests of regular consumers (we call them *honest consumers*) follow a three types of pattern: Zipf-like, exponentially and batch (i.e., generating a specified number of Interests at specified points of simulation) distributions [43]. We also configure the pattern frequency of Interest packets ranging [100..800], where each honest consumer changes five times the frequency randomly. We apply randomly two different replacement policies in PIT table (i.e., perform different actions when limit on number of PIT entries is reached) including LRU and persistent policies. The nodes' cache capacity was randomly set to [100..400] content. We set the low and medium size for cache capacity to evaluate the accuracy and robustness of the proposed ANFIS-based method sufficiently.
The simulation runs over two and a half hours. The collected input-output data pattern during the simulation is divided into three blocks as training (70%) to fit (train) the ANFIS model, the first testing (15%) and the second testing (remaining last 15%) to confirm the ANFIS accuracy after training. During the simulation, honest consumers request content based on the above-mentioned configurations. False-locality (see section 6.2) and locality-disruption (see section 6.3) attacks are issued by adversaries between 0s-30s and 50s-80s, respectively. Finally, the last thirty seconds between 100s-130s, adversaries launch both attacks at the same time to measure the effect of the proposed ANFIS-based cache replacement method in the simultaneous presence of the both attacks.

6.2. False-locality

We consider two types of content poisoning implementation: proactive and active attacks. Firstly, we con-

[2]Reprinted from Computer Networks, Vol 57/16, Mauro Conti, Paolo Gasti and Marco Teoli, A lightweight mechanism for detection of cache pollution attacks in Named Data Networking, pages 3178-3191, Copyright (2013), with permission from Elsevier

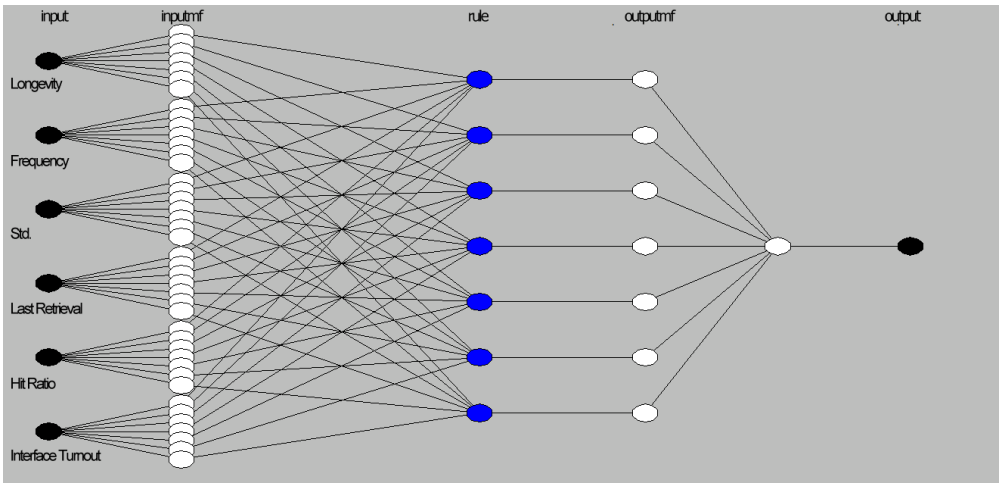
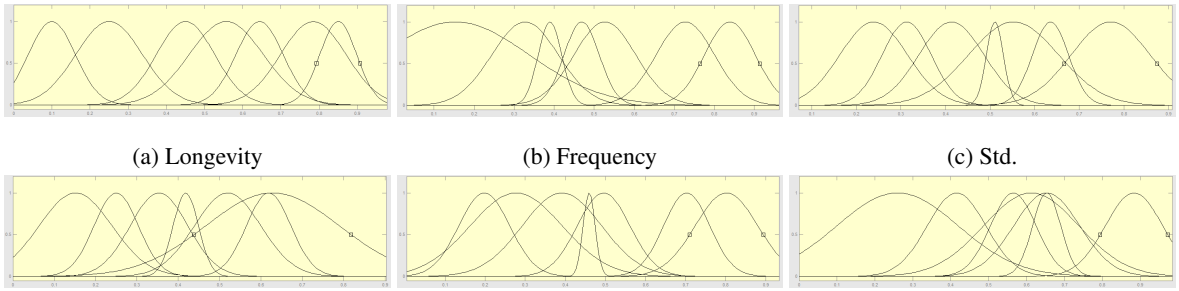


Figure 4: The structure of the proposed ANFIS



(a) Longevity (b) Frequency (c) Std.

(d) Last retrieval (e) Hit ratio (f) Interface Turnout

Figure 5: Final membership functions of the input data

sider a proactive content poisoning attack whereby adversaries anticipated a set of Interest packets for a set of valid content. Adversaries inject fake content into router caches. Assuming a consumer sends an Interest packet which is received by an intermediate router (R_i) and an entry is added to its PIT table. When a router or producer satisfies the Interest packet with a fake content and returns back to the R_i, all the intermediate nodes in the way back as well as R_i are polluted with a fake content. A range of honest consumers are not satisfied if an Interest returns a fake content. After receiving a fake content, they always send the same Interest packet until receive valid content. A range of adversaries behave in the opposite manner. They always ask for bogus content. The adversaries request content according to the uniform distribution. Secondly, we consider an active content poisoning attack whereby adversaries ask some fake content during the simulation run.

We use simulation to measure how many honest consumers can retrieve healthy (valid) content and how fast they can do so when the router caches are poisoned. For proactive scenarios, all routers are pre-populated with different rate of fake content objects, 50%, 80%, and 95% of all the content when the simulation runs. In active scenarios, adversaries request a series of fake content in which intermediate routers are populated with fake content ranging 30%, 50%, and 70% of all the content. We demonstrate that the proposed scheme outperforms the most common policies as Least Frequently Used (LFU) and Least Recently Used (LRU) algorithms in terms of applied performance criteria.

6.3. Locality-disruption

We assume that the adversaries can predict Interest packets from a set of honest consumers to issue Interests for attack purposes. The adversaries can issue Interest packets with 5%, 50%, and 90% of the total number of Interest packets issued by honest consumers according

to the uniform distribution. This allows to explore the effects of low, moderate and high attacks, and whether the proposed ANFIS-based countermeasure is able to identify and mitigate them.

To summarize multiple statistics in the absence and presence of attacks, we define the metric similar to Deng et al. [22] as the key measure of the effectiveness of the attack as:

$$Hit\ damage\ ratio = 1 - \frac{HR(non-attack) - HR(attack)}{HR(non-attack)} \quad (10)$$

Where, $HR(non-attack)$ and $HR(attack)$ denote the hit ratio of honest consumers in the absence/presence of an attack, respectively. When the *Hit damage ratio* is (close to) zero, the attack is completely ineffective, while it is (close to) one, the caching feature is completely under attack. We then demonstrate that the proposed scheme outperforms the most common policies as LFU, LRU independently and in conjunction with CacheShield [17] in terms of applied performance criteria. Xie et al. in [17] introduce CacheShield, a method to shield NDN routers from locality disruption attacks.

7. Experimental results

In this section, we demonstrate through simulations that the proposed ANFIS-based cache replacement method satisfies in a much better way the applied performance criteria as compared to the preexisting methods. Our countermeasure is tested over the two considered topologies in Fig. 3. Each router implements the proposed ANFIS-based technique discussed in Section 5.

7.1. Results of ANFIS design

The training data used for constructing ANFIS model is the obtained statistical data (see section 6) from DFN topology. The constructed ANFIS model is used as a cache replacement method over both XC and DFN topologies in order to test its performance and robustness against cache pollution attacks.

Based on the hybrid training process in ANFIS through the number of constructed cluster centers by fuzzy c-mean clustering, there are seven fuzzy rules. The number of training epochs is 500 and the error tolerance was set to the default value, which is zero. The initial step-size and the increase and decrease rates were set to 0.01, 0.8, and 1.2, respectively. These configuration settings in our application are set up to cover a wide range of learning tasks, which lead to optimization of the training process. Fig. 5 illustrates the final membership functions of input data. To show the efficiency of

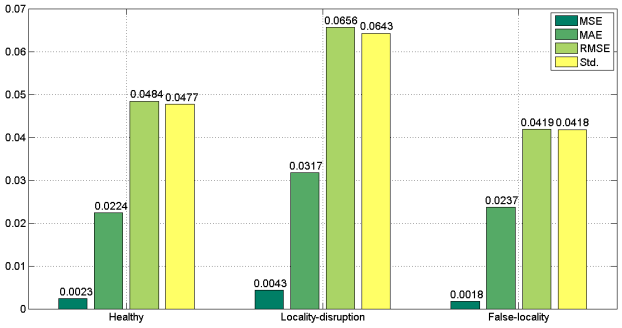


Figure 6: The statistical results on training data set

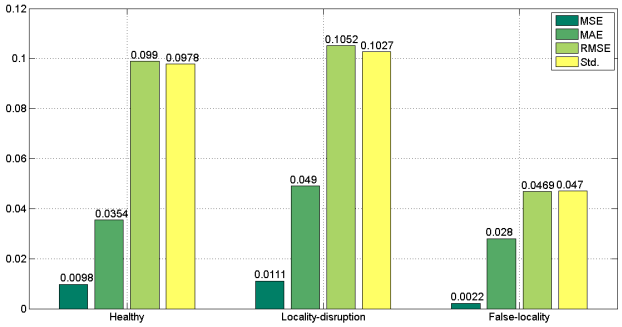


Figure 7: The statistical results on 1st testing data set

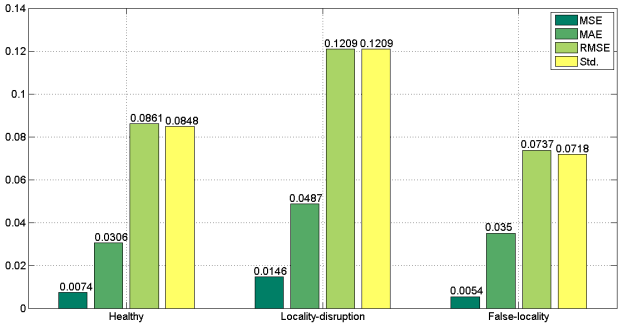
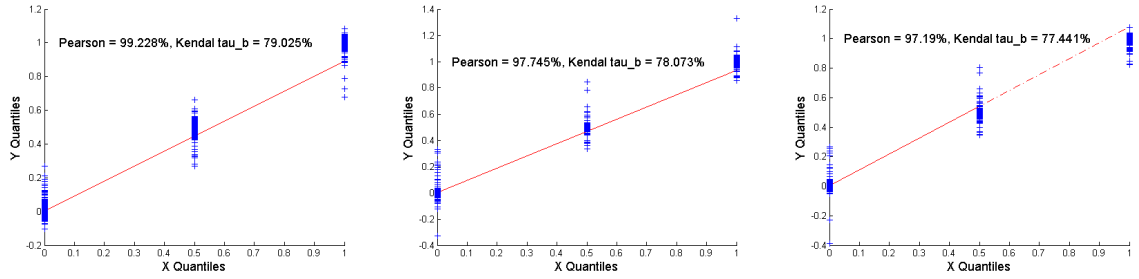


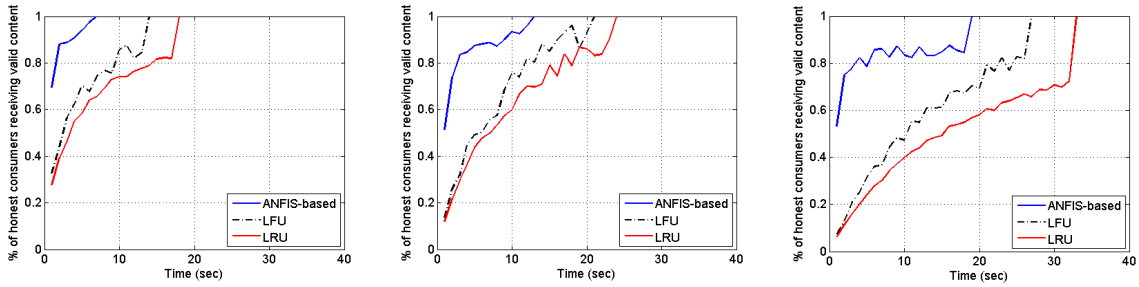
Figure 8: The statistical results on 2nd testing data set

the model, different performance metrics are applied including Mean Square Error (MSE), Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Standard Deviation of the error (Std.), and Quantile-Quantile plot (Q-Q plot) followed by Pearson and Kendall tau_b correlation coefficient divided to training and testing data sets. Numerical results are shown in Figs. 6-9. The plots demonstrate the correspondence between the real values (content type) and corresponding output values predicted by the ANFIS model, indicating that the ANFIS model we have developed is accurate.



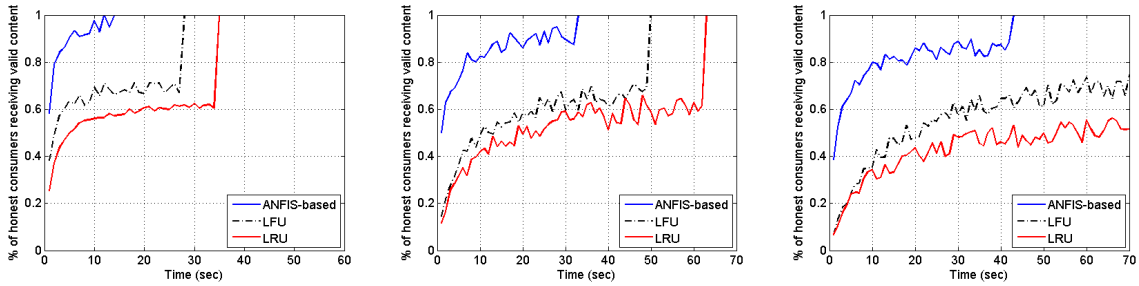
(a) Numerical results on training set (b) Numerical results on 1st testing set (c) Numerical results on 2nd testing set

Figure 9: Q-Q plot and statistical results



(a) 50% pre-populated (b) 80% pre-populated (c) 95% pre-populated

Figure 10: Results of different pre-populated fake content in XC topology (mean of 10 runs)



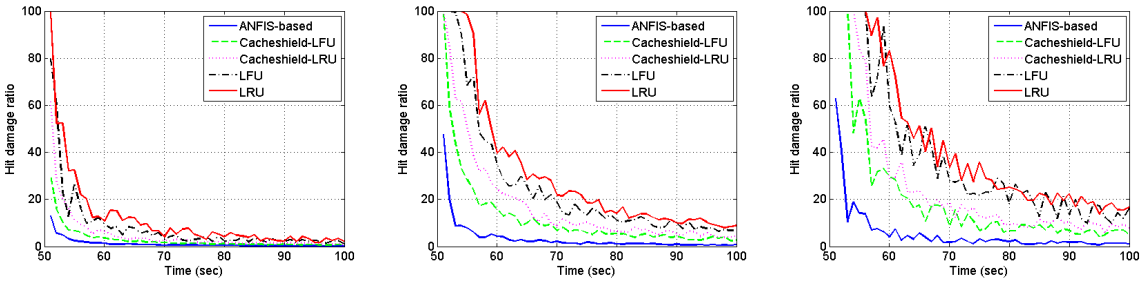
(a) 50% pre-populated (b) 80% pre-populated (c) 95% pre-populated

Figure 11: Results of different pre-populated fake content in DFN topology (mean of 10 runs)

7.2. Mitigating false-locality

We first evaluate the effectiveness of proposed ANFIS-based cache replacement method in a simple network topology using the XC network. Fig. 10 illustrates the average behavior of three methods with different pre-populated fake content rate within 10 runs. The proposed ANFIS-based cache replacement method is more accurate and outperforms other methods in terms of the faster full convergence.

After verifying the correct behavior of ANFIS-based cache replacement method in the XC topology, we consider a more complex network topology using DFN network in Fig. 11. Fig. 11 shows the average of experimental results by ANFIS-based, LRU, and LFU cache replacement methods within 10 runs. As shown in this figure, there is a considerable benefits of the proposed countermeasure implemented by ANFIS model in faster full convergence of the honest consumers. With increased rate of pre-populated fake content, the LRU and LFU methods perform an insignificant behavior in re-



(a) 5% locality-disruption (b) 50% locality-disruption (c) 90% locality-disruption

Figure 12: Results of Hit damage ratio for locality-disruption attack in XC topology (mean of 10 runs)

moving the fake content from the caches. Whereas, the proposed ANFIS-based method performs more accurate and efficient in removing the fake content from the caches and satisfies all the honest consumers in a timely manner.

7.3. Mitigating locality-disruption

Simulation results in Figs. 12 and 13 show that our cache replacement technique can quickly detect the content placed with the goal of performing locality-disruption attacks and replace them when a new content is added to a full cache. These figures show that routers using ANFIS-based cache replacement method successfully outperforms four applied cache replacement algorithms in a timely manner. The most stunning result is the extreme vulnerability of the LRU and the LFU to pollution attacks.

The experimental results in Figs. 12a-12c and 13a-13c indicate that the ANFIS-based cache replacement technique is more resilient than the preexisting methods against locality-disruption attacks. Despite the fact that the hit damage ratio is still quite high by ANFIS-based technique in the early times of the simulation, the application of the ANFIS-based technique is quite effective and more reliable against low, middle, and high rate pollution attacks.

7.4. Mitigating combination of both attacks at the same time

Adversaries can launch both false-locality and locality-disruption attacks at the same time. For instance, the same set of attackers can launch false-locality attacks by pre-populating 50% of the total honest consumers' Interest requests, and at the same time they start locality-disruption attacks to interfere the content locality by requesting the rest 50% of the honest consumers' Interest requests in the caches.

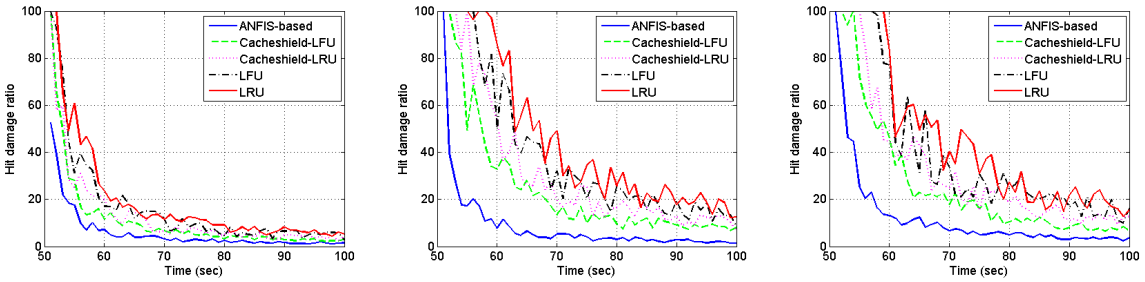
According to the proposed ANFIS-based cache replacement method discussed in section 5, the existence of locality-disruption attacks will not affect the detection of false-locality attacks and vise versa. First, the proposed method tries to detect false-locality attacks by assigning goodness value close to zero, and once detected, they are removed from the caches. Then those content with the goodness value close to 0.5, detected as locality-disruption attacks, would be replaced separately when a new content enters and cache space is full.

We vary the behavior of attackers for executing false-locality and locality-disruption attacks between 100 and 130 seconds of the simulation run. Figs. 14-16 and 17-19 are shown the results of mitigating both cache pollution attacks at the same time with different strategies in XC and DFN topologies, respectively. Experimental results demonstrate that our proposed method is more resilient and more accurate than preexisting methods to the mixture of attacks. The most stunning result is the extreme vulnerability of the LRU and LFU algorithms to the active false-locality attacks as compared to the proactive false-locality attacks. Thus, the proposed ANFIS-based cache replacement mechanism in the considered simulation environments offers visibly promising performance in presence of cache pollution attacks.

7.5. The overhead cost

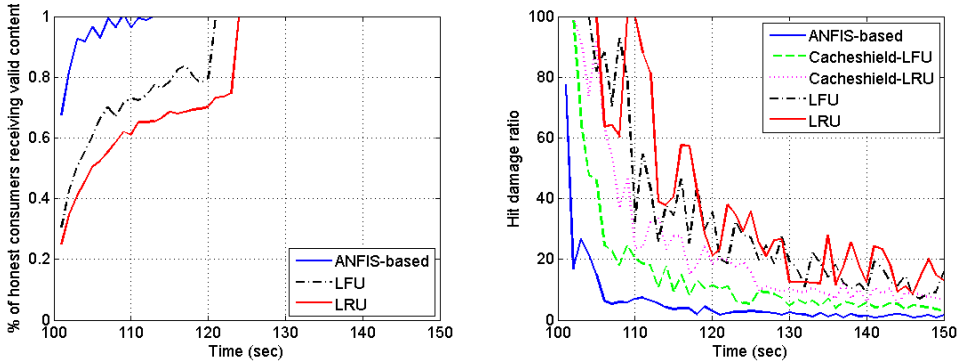
In this section, we assess the overhead cost of our proposed method and preexisting schemes in presence of adversaries. In particular, we are interested in determining the overhead of the average number of arrival data packets for legitimate users in routers and the operation overhead of the methods.

1. The overhead of the average of arrival data packets: It guarantees that this amount of data packet was actually transferred over the channel during the cache



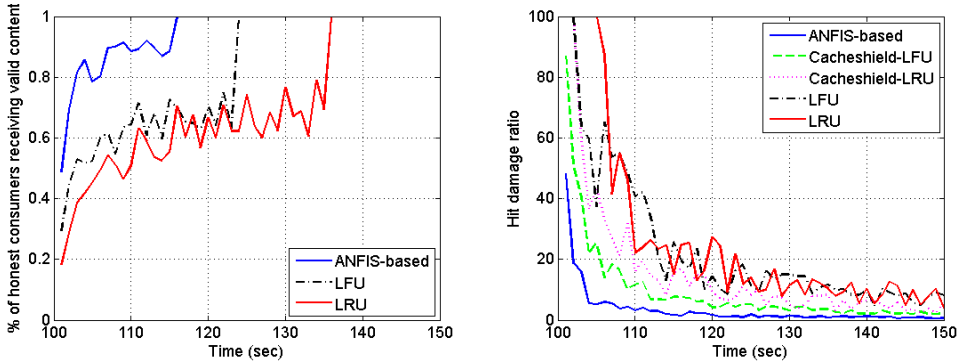
(a) 5% locality-disruption (b) 50% locality-disruption (c) 90% locality-disruption

Figure 13: Results of Hit damage ratio for locality-disruption attack in DFN topology (mean of 10 runs)



(a) Receiving valid content by 30% false-locality (b) The hit damage ratio by 70% locality-disruption

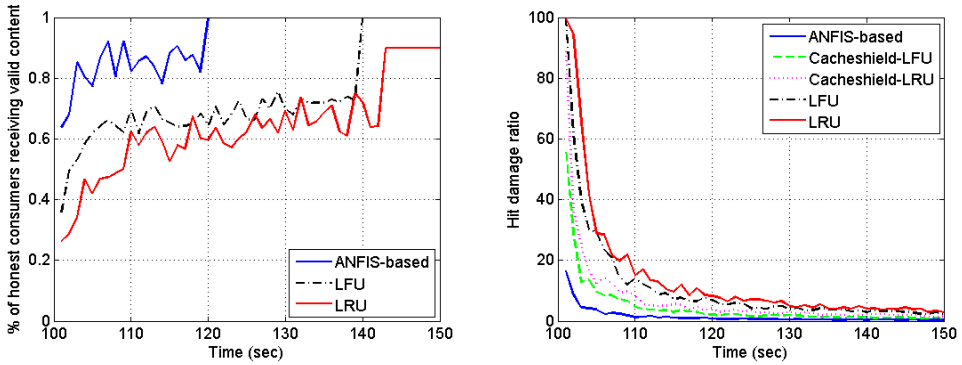
Figure 14: The results for 30% false-locality and 70% locality-disruption in XC topology (mean of 10 runs)



(a) Receiving valid content by 50% false-locality (b) The hit damage ratio by 50% locality-disruption

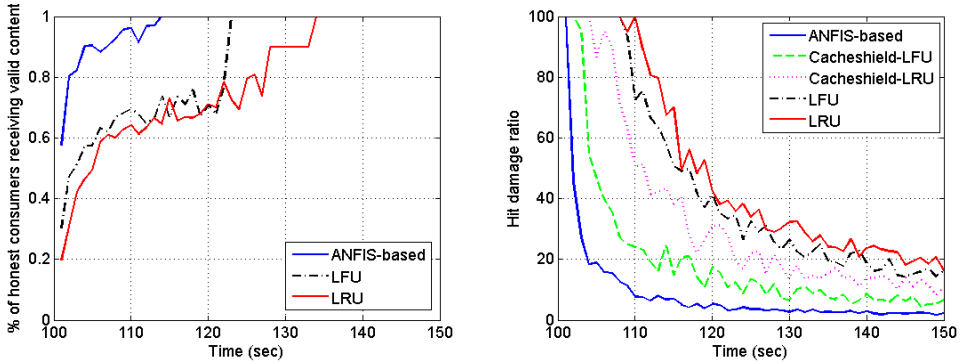
Figure 15: The results for 50% false-locality and 50% locality-disruption in XC topology (mean of 10 runs)

pollution attacks. Figs. 20 and 21 show the average of overhead of transmitted data packets in routers in the XC and DFN networks, respectively. We can observe that the our proposed method outperforms other methods based on the lower overhead of data transmission. Our results confirm that the most data packets were able to cache to the closest edge routers (i.e., close routers to the legitimate consumers) by mitigating effectively both



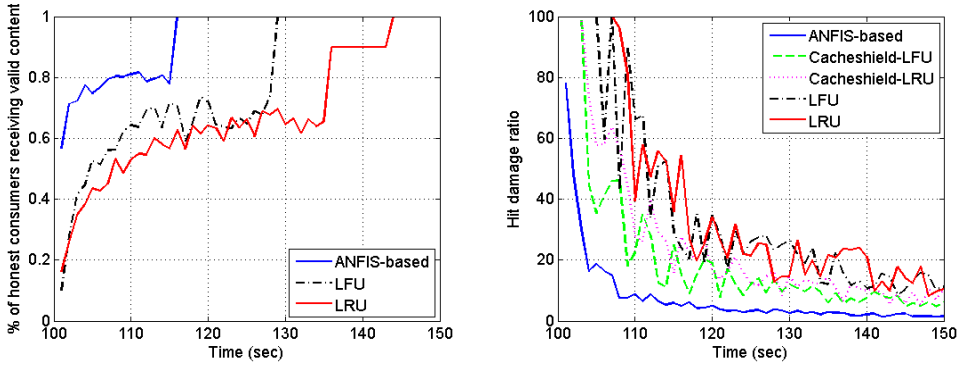
(a) Receiving valid content by 70% false-locality (b) The hit damage ratio by 30% locality-disruption

Figure 16: The results for 70% false-locality and 30% locality-disruption in XC topology (mean of 10 runs)



(a) Receiving valid content by 30% false-locality (b) The hit damage ratio by 70% locality-disruption

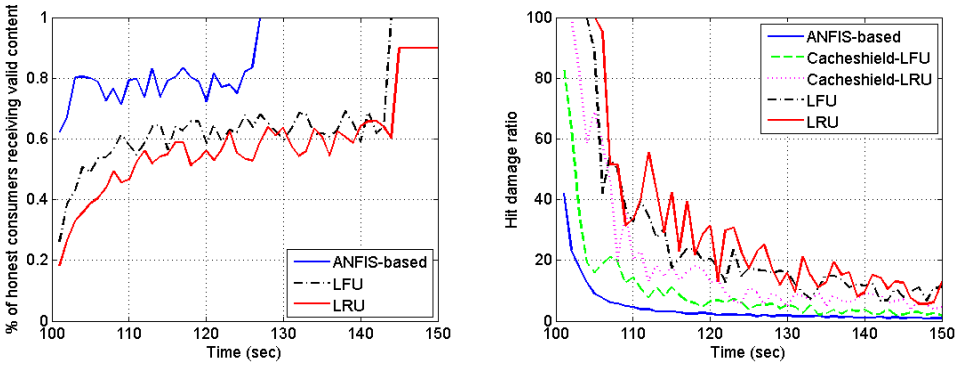
Figure 17: The results for 30% false-locality and 70% locality-disruption in DFN topology (mean of 10 runs)



(a) Receiving valid content by 50% false-locality (b) The hit damage ratio by 50% locality-disruption

Figure 18: The results for 50% false-locality and 50% locality-disruption in DFN topology (mean of 10 runs)

attacks. Our results also show that the overhead of transmitting data packets by LRU and LFU algorithms are greater than our proposed method and the CacheShield, making the attack more effective.



(a) Receiving valid content by 70% false-locality (b) The hit damage ratio by 30% locality-disruption

Figure 19: The results for 70% false-locality and 30% locality-disruption in DFN topology (mean of 10 runs)

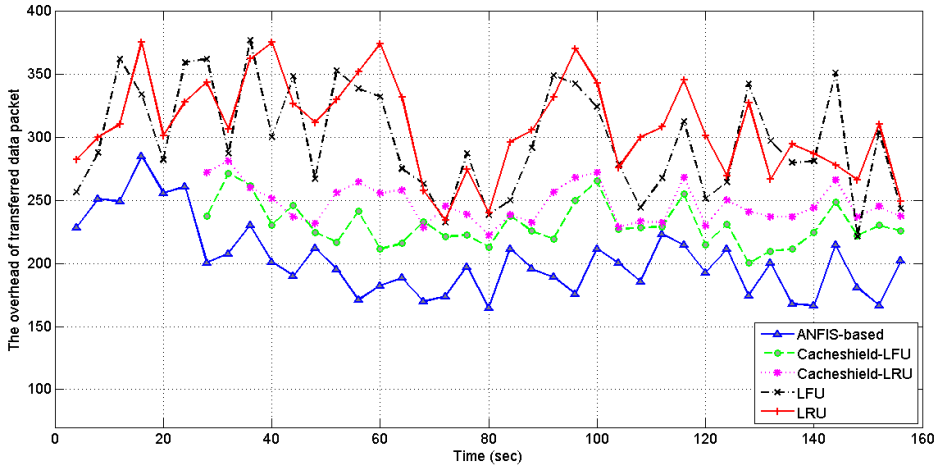


Figure 20: The average of arrival data packets in XC topology

2. The operation overhead: This is the amount of processing time to execute the caching algorithms within operating system. Table 1 shows that the proposed method seems to be less time consuming than the other methods except LRU and LFU algorithms when attacks do not run simultaneously. The results in Table 1 indicate that the proposed approach can improve the performance as compared to LRU and LFU algorithms in terms of the operation overhead up to 3.93% and 2.15 %, and 4.15% and 3.78% in XC and DFN topologies respectively, when both cache pollution attacks are simultaneously implemented. According to the obtained results, by increasing rate of attacks, the overhead of our proposed method is considerably decreased as compared to LRU and LFU. The results from Table 1 also confirm that the our proposed method outperforms sufficiently the CacheShield-LRU and CacheShield-LFU methods in terms of the operation overhead up to 11.56% and 13.79%, and 18.67% and 20.14% in XC and DFN, respectively.

To evaluate the effectiveness and efficiency of the proposed method, we illustrate that the proposed ANFIS-based method provides a suitable compromise between overhead (i.e., the overhead of the arrival data packets in Figs. 20 and 21, and the operation overhead of the algorithms in Table 1) and applied performance metrics including the percentage of legitimate consumers receiving valid content (Figs. 10-11 and 14a-19a) and the hit damage ratio (Figs. 12-13 and 14b-19b) as compared to common existing countermeasures. Therefore,

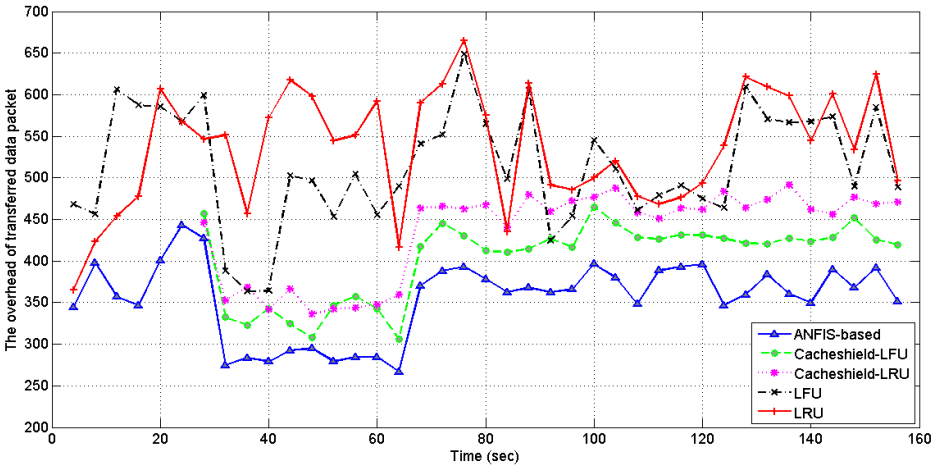


Figure 21: The average of arrival data packets in DFN topology

Table 1: Comparing operation overhead achieved by the proposed scheme over other methods (mean of 10 runs)

Time (sec)	Attack	Percent of worsening (↓) and improving (↑) (%)			
		LRU	LFU	CacheShield-LRU	CacheShield-LFU
XC topology:					
0-50 (false-locality attacks)	50%	↓ 8.83	↓ 7.98	-	-
	80%	↓ 7.11	↓ 4.36	-	-
	95%	↓ 5.32	↓ 2.81	-	-
50-100 (locality-disruption attacks)	5%	↓ 8.62	↓ 7.37	↑ 11.41	↑ 11.51
	50%	↓ 8.47	↓ 6.29	↑ 10.11	↑ 11.24
	90%	↓ 6.83	↓ 3.98	↑ 11.28	↑ 12.76
100-150 (combination of both attacks)	30-70%	↑ 1.74	↑ 1.31	↑ 10.18	↑ 13.07
	50-50%	↑ 2.59	↑ 1.64	↑ 11.56	↑ 12.34
	70-30%	↑ 3.93	↑ 2.15	↑ 11.01	↑ 13.79
DFN topology:					
0-50 (false-locality attacks)	50%	↓ 9.52	↓ 7.45	-	-
	80%	↓ 8.17	↓ 6.43	-	-
	95%	↓ 8.03	↓ 5.14	-	-
50-100 (locality-disruption attacks)	5%	↓ 9.21	↓ 9.33	↑ 16.73	↑ 15.33
	50%	↓ 9.01	↓ 7.75	↑ 17.11	↑ 16.03
	90%	↓ 6.83	↓ 7.24	↑ 15.91	↑ 17.76
100-150 (combination of both attacks)	30-70%	↑ 1.42	↑ 1.66	↑ 17.34	↑ 20.02
	50-50%	↑ 2.84	↑ 2.03	↑ 17.13	↑ 19.38
	70-30%	↑ 4.15	↑ 3.78	↑ 18.67	↑ 20.14

the extensive analysis satisfies the objectives of the experiment in terms of the applied performance metric and ensure that the proposed ANFIS-based caching for mitigating cache pollution attacks in NDN can yield high accuracy as compared to other methods without very much computational cost.

8. Conclusion

In this paper, we proposed a novel ANFIS-based cache replacement method to mitigate two generic cache pollution attacks namely false-locality and locality-disruption in NDN. Simulation results showed that the proposed method provides very accurate results as compared to LRU and LFU algorithms independently and in conjunction with CacheShield scheme. Experimental results and analysis show the proposed ANFIS-based cache replacement method is very effective in determining and mitigating the fake content, and has a very high detection rate of locality-disruption attacks to replace them when new content is added to a full cache in a timely manner. The extensive analysis satisfies the

objectives of the experiment and ensure that the proposed ANFIS-based caching for mitigating cache pollution attacks can yield high accuracy as compared to other methods without very much computational cost. Future work includes devising several improvements to the approach presented in this paper and its use in larger and more complex network topologies.

9. Acknowledgments

This work was partially supported by projects TIN2013-47272-C2-2 and SGR-2014-881.

References

[1] Jacobson, V., Smetters, D.K., Thornton, J.D., Plass, M.F., Briggs, N.H., Braynard, R.L.. Networking named content. In: Proceedings of the 5th international conference on Emerging networking experiments and technologies. ACM; 2009, p. 1 – 12.

[2] Zhang, G., Li, Y., Lin, T.. Caching in information centric networking: A survey. Computer Networks 2013;57(16):3128 – 3141.

[3] Sourlas, V., Flegkas, P., Tassiulas, L.. A novel cache aware routing scheme for information-centric networks. Computer Networks 2014;59:44 – 61.

[4] Karami, A., Guerrero-Zapata, M.. A fuzzy anomaly detection system based on hybrid pso-kmeans algorithm in content-centric networks. Neurocomputing 2015;149(Part C):1253 – 1269.

[5] qing Wang, G., Huang, T., Liu, J., ya Chen, J., jie Liu, Y.. Modeling in-network caching and bandwidth sharing performance in information-centric networking. The Journal of China Universities of Posts and Telecommunications 2013;20(2):99 – 105.

[6] Dannewitz, C., Kutscher, D., Ohlman, B., Farrell, S., Ahlgren, B., Karl, H.. Network of information (netinf) an information-centric networking architecture. Computer Communications 2013;36(7):721 – 735.

[7] Zhang, L., Estrin, D., Burke, J., Jacobson, V., Thornton, J., Smetters, D.K., et al. Named data networking (ndn) project. Tech. Rep. PARC TR-2010-3; Palo Alto Research Center; 2010.

[8] Karami, A., Guerrero-Zapata, M.. A hybrid multiobjective rbf-pso method for mitigating dos attacks in named data networking. Neurocomputing 2015;151(Part 3):1262 – 1282.

[9] Chaabane, A., Cristofaro, E.D., Kaafar, M.A., Uzun, E.. Privacy in content-oriented networking: Threats and countermeasures. ACM SIGCOMM Computer Communication Review 2013;43(3):25 – 33.

[10] Acs, G., Conti, M., Gasti, P., Ghali, C., Tsudik, G.. Cache privacy in named-data networking. In: 33rd IEEE International Conference on Distributed Computing Systems (ICDCS). 2013, p. 41 – 51.

[11] Lee, H., Nakao, A.. User-assisted in-network caching in information-centric networking. Computer Networks 2013;57(16):3142 – 3153.

[12] Kim, Y., Yeom, I.. Performance analysis of in-network caching for content-centric networking. Computer Networks 2013;57(13):2465 – 2482.

[13] Carofiglio, G., Gallo, M., Muscariello, L.. On the performance of bandwidth and storage sharing in information-centric networks. Computer Networks 2013;57(17):3743 – 3758.

[14] Chai, W.K., He, D., Psaras, I., Pavlou, G.. Cache less for more in information-centric networks (extended version). Computer Communications 2013;36(7):758 – 770.

[15] Xylomenos, G., Ververidis, C.N., Siris, V.A., Fotiou, N., Tsilopoulos, C., Vasilakos, X., et al. A survey of information-centric networking research. IEEE Communications Surveys Tutorials 2013;:1 – 26.

[16] Lauinger, T., Laoutaris, N., Rodriguez, P., Strufe, T., Biersack, E., Kirda, E.. Privacy risks in named data networking: what is the cost of performance? ACM SIGCOMM Computer Communication Review 2012;42(5):54 – 57.

[17] Xie, M., Widjaja, I., Wang, H.. Enhancing cache robustness for content-centric networking. In: IEEE Proceedings on INFOCOM. 2012, p. 2426 – 2434.

[18] Park, H., Widjaja, I., Lee, H.. Detection of cache pollution attacks using randomness checks. In: IEEE International Conference on Communications (ICC). 2012, p. 1096 – 1100.

[19] Ghali, C., Tsudik, G., Uzun, E.. Needle in a haystack: Mitigating content poisoning in named-data networking. In: Proceedings of NDSS Workshop on Security of Emerging Networking Technologies (SENT). 2014,.

[20] Conti, M., Gasti, P., Teoli, M.. A lightweight mechanism for detection of cache pollution attacks in named data networking. Computer Networks 2013;57(16):3178 – 3191.

[21] Chen, H., Xiao, Y., Vrbsky, S.V.. An update-based step-wise optimal cache replacement for wireless data access. Computer Networks 2013;57(1):197 – 212.

[22] Deng, L., Gao, Y., Chen, Y., Kuzmanovic, A.. Pollution attacks and defenses for internet caching systems. Computer Networks 2008;52(5):935–956.

[23] Kaya, C.C., Zhang, G., Tan, Y., Mookerjee, V.S.. An admission-control technique for delay reduction in proxy caching. Decision Support Systems 2009;46(2):594 – 603.

[24] Romano, S., ElAarag, H.. A neural network proxy cache replacement strategy and its implementation in the squid proxy server. Neural Computing and Applications 2011;20(1):59 – 78.

[25] Ahmed, W.A., Shamsuddin, S.M.. Neuro-fuzzy system in partitioned client-side web cache. Expert Systems with Applications 2011;38:14715 – 14725.

[26] Bagheri, A., Peyhani, H.M., Akbari, M.. Financial forecasting using anfis networks with quantum-behaved particle swarm optimization. Expert Systems With Applications 2014;41(14):6235 – 6250.

[27] Budyal, V., Manvi, S.. Anfis and agent based bandwidth and delay aware anycast routing in mobile ad hoc networks. Journal of Network and Computer Applications 2014;39:140 – 151.

[28] Güneri, A.F., Ertay, T., Yücel, A.. An approach based on anfis input selection and modeling for supplier selection problem. Expert Systems With Applications 2011;38:14907 – 14917.

[29] Moayer, S., Bahri, P.A.. Hybrid intelligent scenario generator for business strategic planning by using anfis. Expert Systems With Applications 2009;36:7729 – 7737.

[30] Guillaume, S.. Designing fuzzy inference systems from data: an interpretability-oriented review. IEEE Transactions on Fuzzy Systems 2001;9(3):426 – 443.

[31] Naderloo, L., Alimardani, R., Omid, M., Sarmadian, F., Javadikia, P., Torabi, M.Y., et al. Application of {ANFIS} to predict crop yield based on different energy inputs. Measurement 2012;45(6):1406 – 1413.

[32] Compagno, A., Conti, M., Gasti, P., Tsudik, G.. Poseidon: Mitigating interest flooding ddos attacks in named data networking. In: 38th Annual IEEE Conference on Local Computer Networks (LCN). 2013, p. 630 – 638.

[33] Jang, J.S.. Anfis: adaptive-network-based fuzzy inference system. IEEE Transactions on Systems, Man and Cybernetics

1993;23(3):665 – 685.

[34] Singh, R., Kainthola, A., Singh, T.. Estimation of elastic constant of rocks using an {ANFIS} approach. Applied Soft Computing 2012;12(1):40 – 45.

[35] Jiang, H., Kwong, C., Ip, W., Wong, T.. Modeling customer satisfaction for new product development using a pso-based {ANFIS} approach. Applied Soft Computing 2012;12(2):726 – 734.

[36] Jang, J.S.. Neuro-fuzzy modeling: Architectures, analyses, and applications. Ph.D. thesis; University of California, Berkeley; 1992.

[37] Vakali, A.. Evolutionary techniques for web caching. Distributed and Parallel Databases 2002;11(1):93 – 116.

[38] Gao, Y., Deng, L., Kuzmanovic, A., Chen, Y.. Internet cache pollution attacks and countermeasures. In: Proceedings of the 14th IEEE International Conference on Network Protocols (ICNP). 2006, p. 54 – 64.

[39] Shanbhag, S.. Design and implementation of parallel anomaly detection. 2007.

[40] Karami, A., Guerrero-Zapata, M.. Mining and visualizing uncertain data objects and named data networking traffics by fuzzy self-organizing map. In: In Proceedings of the Second International Workshop on Artificial Intelligence and Cognition (AIC); vol. 1315. 2014, p. 156 – 163.

[41] Karami, A., Johansson, R.. Utilization of multi attribute decision making techniques to integrate automatic and manual ranking of options. Journal of Information Science and Engineering 2014;30(2):519 – 534.

[42] Jin, Y.. Multi-objective machine learning; vol. 16. Springer; 2006.

[43] Afanasyev, A., Moiseenko, I., Zhang, L.. ndnSIM: NDN simulator for NS-3. Technical Report NDN-0005; NDN; 2012. URL `http://named-data.net/techreports.html`.

[44] Heckmann, O., Piringer, M., Schmitt, J., Steinmetz, R.. On realistic network topologies for simulation. In: Proceedings of the ACM SIGCOMM Workshop on Models, Methods and Tools for Reproducible Network Research. MoMeTools '03; New York, NY, USA: ACM; 2003, p. 28 – 32.