Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 159 (2019) 1182–1189

Procedia Computer Science

www.elsevier.com/locate/procedia

23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Determination of the turning point of cache efficiency in computer networks with logic Eτ

Avelino Palma Pimenta Junior[b], Jair Minoro Abe[a,b,*]

[a]*Institute For Advanced Studies, University of São Paulo, Brazil*
[b]*Paulista University, Graduate Program in Production Engineering, R. Dr. Bacelar 1212, 04026-002 São Paulo, Brazil*

Abstract

Object caches are used to minimize data traffic in various areas of Information Technology, including computer networks. In this scenario, they are usually hosted in proxies, storing page objects (texts, figures, among others), and implementing access control policies. Its correct operation can provide a significant gain of performance in data exchange since it allows the immediate response of requested resources. This study aims to discuss the different states of the efficiency of two distinct types of computer network caches and determine the changing dynamics of these states with Logic Eτ.

Keywords: Type your keywords here, separated by semicolons ;

1. Introduction

The task to analyze the efficiency of a computer network cache will be based on Logic Eτ, composed of memory and disk cache. The hit rates for each cache type will be evaluated over a given period and will formulate hypotheses

* Corresponding author. Tel.: ++55 11 5586 4145
E-mail address: jair.abe@docente.unip.br

10.1016/j.procs.2019.09.287

about any changes in its state.

Computer networks can be considered one of the main ways of transmitting information, if not the main one. Monitoring the information has become a critical factor in technology sectors [1] and is part of the backbone of Information Technology in various educational institutions and companies. A computer network consists of several connected hosts, which can be represented by a desktop, a laptop, a smartphone, wearable devices, biomedical sensors, among others [2][3]. In such a heterogeneous client environment, dynamic content adaptation and delivery services are becoming an essential requirement for the new Internet service infrastructure [4].

This paper is organized as follows: in Section 2, the methodology and a comparison of the evaluated project with existing anomaly detection models are discussed; in Section 3, the evaluated network attributes used are presented; in Section 4, the basic concepts of Logic Eτ are introduced; section 5 discusses the development of the project considering the network attributes and Logic Eτ; section 6 presents the results and discussion, and the conclusion is found in Section 7.

2. Methodology and comparison with existing models

In order to compare with existing models, it is necessary to present the materials and tools used in this project. The information was acquired from a proxy server, representing a gateway for external networks. The evaluated network is used by the academic staff of an educational institution, exclusively by its students and teachers.

The network attributes were normalized, and concepts of the Logic Eτ were applied, considering favorable and contrary evidence for each scenario.

Unlike [5] and [6], which rely on simulated data to emulate a real network environment and the synthetic generation of anomalies, this project employs continuously gathered data from the operation of a real working network for the learning process. Another difference from [7] is that this project uses Digital Signature of Network Segment using Flow Analysis (DSNSF), which establishes a profile for the normal behavior of a network segment by considering the history of its movement. A possible problem: when a real-time system is not considered for this kind of task, any changes in the network layout or its availability may impact the analyzer learning process, since the history may not represent the actual state of the network.

Another significant difference from the work of [8], in which data traffic was used as an analytical measure, without distinction of individual attributes that could represent different operating situations of the network, is that, in this project, the network attributes are treated individually.

3. Evaluated network attributes

As mentioned by [9]: "The amount of information that travels across the Internet has increased dramatically over the past decades because of the huge growth in the number of internet users." Also, the growing number of services and applications, as well as the many advances in Information Technology, make networks and information systems essential for the survival of all educational enterprises, organizations, and institutions [10]. The growth of the global computer network also leads to an increase in the complexity of its infrastructure. Thus, the classical methods of network analysis may not be the adequate ones for this scenario[11].

Computer networks use routers to communicate with each other. As mentioned by [12], "Routing is the process of sending data packets from the host of origin to the destination host, which is performed by the routers." Some elements may be interesting for network traffic analysis. The following analyzable attributes were considered:

- TCP_MEM_HIT/200
- TCP_HIT/200

The first attribute (TCP_MEM_HIT/200) represents the requests that had a positive response in the search for objects stored in memory. The second attribute (TCP_HIT/200) represents the response for objects stored in the disk.

4. The Logic Eτ

There are high levels of uncertainty when monitoring data in computer networks. The argument for this assertion is based on the principle that user actions are presented as random elements [13]. Therefore, the use of a non-classical logic becomes an option. Logic Eτ can be a viable technique to search for indications of problems during the regular operation of the network or by intentional elements [14] [15]. In the latter case, the problems may be caused by misuse or malicious software [16].

According to [17]: "The atomic formulas of the Logic Eτ are the type p(μ, λ), where $(\mu, \lambda) \in [0, 1]^2$ ([0, 1] is the real unit interval) and p denote a propositional variable". Therefore, among several readings, p(μ, λ) can be intuitively read: "It is assumed that the favorable evidence of p is μ and the contrary evidence of p is λ." Thus, we have, for instance, the following particular readings:

- $p_{(1.0,\ 0.0)}$ can be read as a true proposition
- $p_{(0.0,\ 1.0)}$ as false
- $p_{(1.0,\ 1.0)}$ as inconsistent
- $p_{(0.0,\ 0.0)}$ as paracomplete, and
- $p_{(0.5,\ 0.5\ )}$ as an indefinite proposition

The uncertainty and certainty degrees associated with (μ, λ) are defined [18][19]:

- Uncertainty Degree: $G_{un}(\mu, \lambda) = \mu + \lambda - 1$ $(0 \leq \mu, \lambda \leq 1)$;
- Certainty Degree: $G_{ce}(\mu, \lambda) = \mu - \lambda$ $(0 \leq \mu, \lambda \leq 1)$;

An order relation is defined on $[0, 1]^2$: $(\mu_1, \lambda_1) \leq (\mu_2\ \lambda_2) \Leftrightarrow \mu_1 \leq \mu_2$ and $\lambda_2 \leq \lambda_{1,}$ forming a lattice, which is symbolized by τ.

With the degrees of certainty and uncertainty, one can determine the following 12 output states, shown in Table 1:

Table 1. Extreme and the Non-Extreme States

Extreme states	Symbol	Non-extreme states	Symbol
True	V	Quasi-true tending to Inconsistent	QV→T
False	F	Quasi-true tending to Paracomplete	QV→⊥
Inconsistent	T	Quasi-false tending to Inconsistent	QF→T
Paracomplete	⊥	Quasi-false tending to Paracomplete	QF→⊥
		Quasi-inconsistent tending to True	QT→V
		Quasi-inconsistent tending to False	QT→F
		Quasi-paracomplete tending to True	Q⊥→V
		Quasi-paracomplete tending to False	Q⊥→F

Extreme and non-extreme states are shown in Fig. 1:

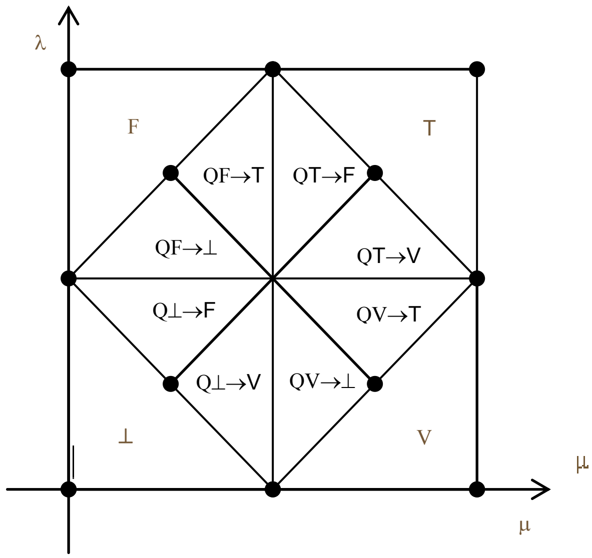

Figure 1: Extreme and non-extreme states of the Lattice τ

In Fig 2, the states, along with certainty and uncertainty degrees, are shown, as well as the control values.

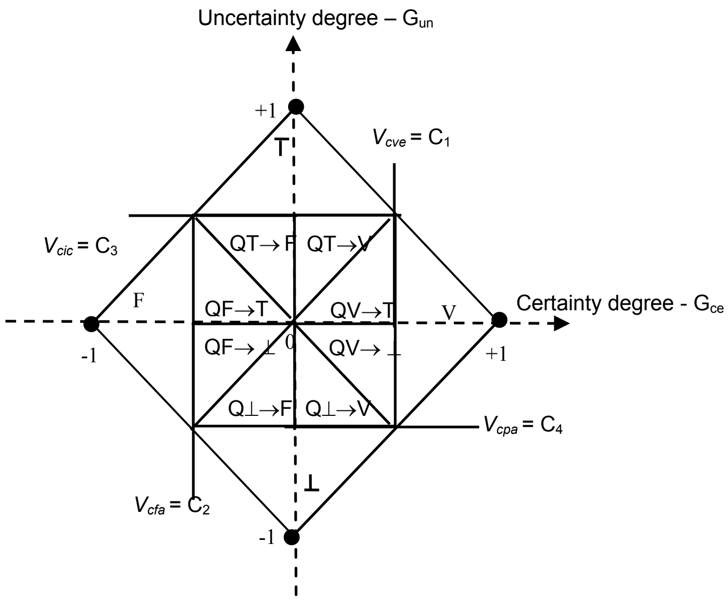

Figure 2: Certainty / Uncertainty degrees with decision states of the Lattice

5. Development

The cache performance analysis has considered an entire period of local network operation, from 8 AM to 10 PM. For this analysis, there are no influences from the cache created on the previous day. Each operation cycle is reset, i.e., they are deleted for the creation of a new one. This operation is performed automatically at the end of the previous day's operation.

The research was based on data obtained in a computer network of a public educational institution, composed by approximately 230 hosts, and it covers not only the students and professors of this institution (academic area) but also the employees (administrative area). Therefore, they can be described as sectors with different demands and behaviors.

The fundamental element in this analysis is represented by the accesses made by the network users, that is, the situations in which the users have searched for some resource in the network. Usually, it is difficult to predict the actions of the users, who usually present a random behavior pattern in the network. Therefore, the cache structuring presents an even greater element of uncertainty, which must be considered.

The number of accesses throughout the day can be represented by the "Total Requisition" field, which is counted at every hour of operation of a computer network. The access values can be represented by Fig. 3:

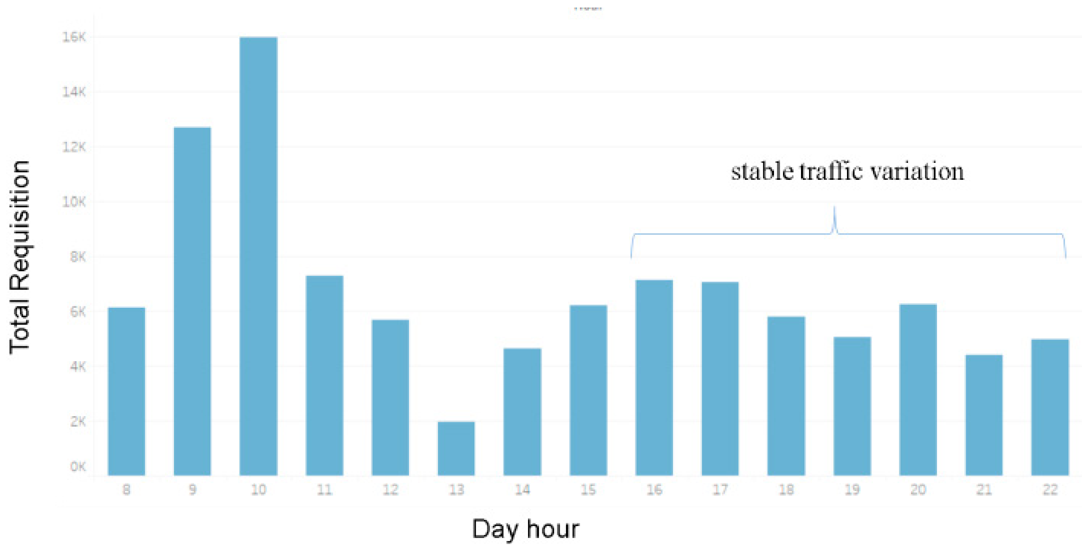

Figure 3: “Total Requisition” values throughout the day operation

Similarly, a representation of the cache hits, considering the data of memory and disk, can be represented by figure 4:

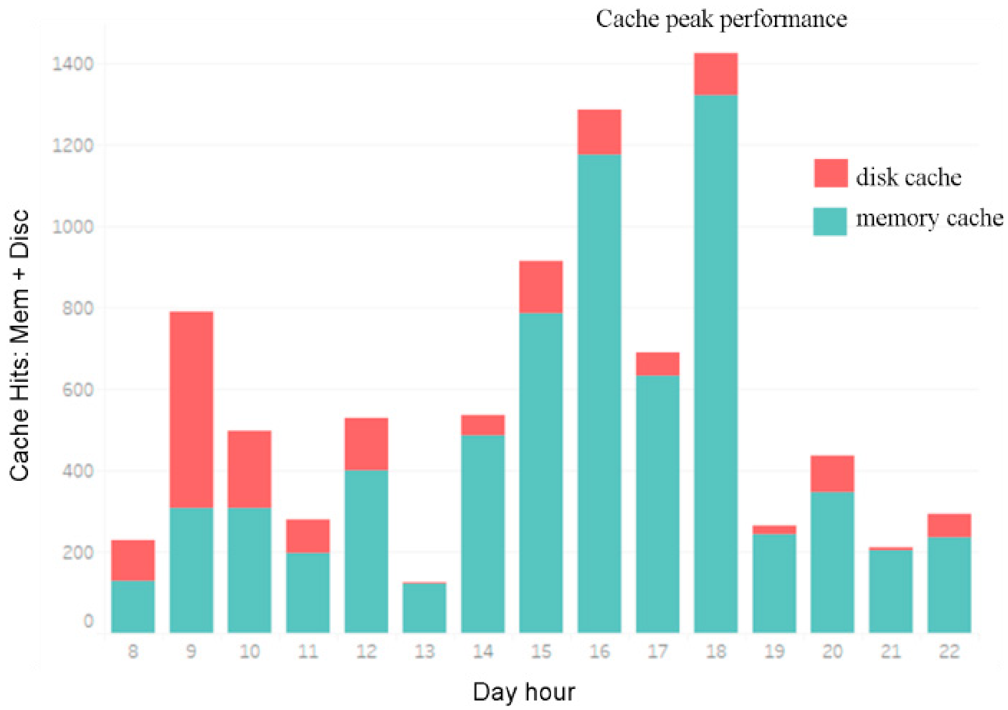

Figure 4: Memory and disk cache hits

The cache obtained its best performance starting at 15:00, presenting a tendency of continuous growth until 18:00 when it reaches its peak performance (Figure 4). Then, it suffers a sudden performance drop, remaining at low levels from 19:00 to 22:00. On the other hand, we can verify, in Figure 3, that the "Total Requisition" field remained stable starting at 15:00, with little variation compared to the cache, hits performance.

6. Results and Discussion

From the values normalized above, the Logic Eτ was applied to verify in which state this abrupt variation could be framed. As favorable evidence, the values of hits in-memory cache (TCP_MEM_HIT/200) were applied. As contrary evidence, there was the application of the disk cache hits (TCP_HIT/200). It is notorious that access to resources in the memory is significantly faster than on disk, which allows a more effective response to the requests. The Lattice τ was generated by presenting the various analyzed cache hits, represented by Fig. 5:

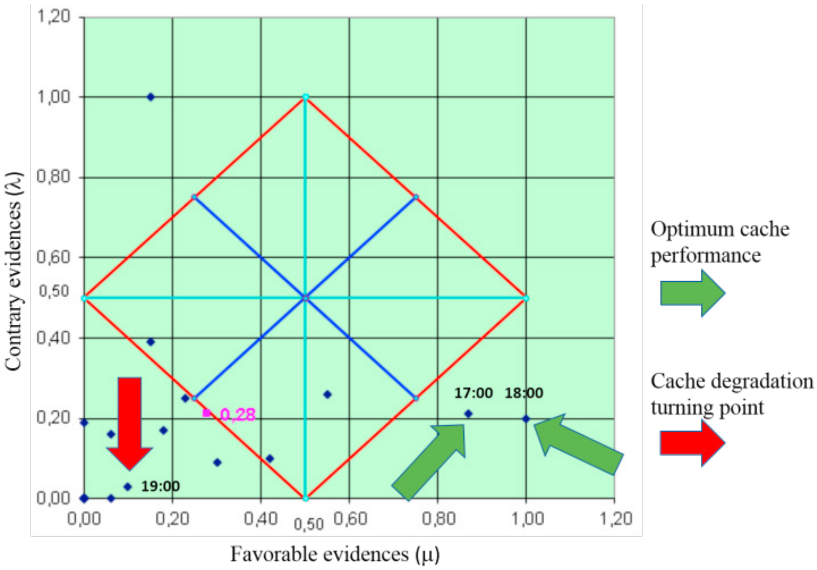

Figure 5: Favourable and contrary evidence from cache hits on Lattice τ

From the presented states in the lattice described in Figure 5, the only two points in which the cache operation can be considered efficient are represented by the intervals of 17:00 and 18:00. Next, there is a sudden loss of performance starting at 19:00, when both favorable and contrary evidence decline.

In comparison with the attribute "Total Requisitions," it is observed that there is no increase, at any moment, in traffic data that could justify this significant change. Instead, data decrease is observed in the network.

Therefore, after achieving its best performance, both disk and memory caches were no longer able to handle the requests of stored objects. Such a phenomenon may be explained by the fact that, after 18:00, there may have been a significant change in the object profile requested by the users. Considering that it consists of an academic network, a change of context must be evaluated, considering that the interval comprises the entrance of the night shift students.

7. Conclusion

With this study, it has been possible not only to determine the maximum performance in absolute values of the network cache but also to quantify its respective components.

Also, it has been possible to determine that the leading cause of the performance loss of the cache was much more due to the use of the network by the students, than in comparison to the institution employees, who presented relatively constant behavior. By the time students start the night shift, the cache has its performance decreased quickly, and it does not have enough time to recover its past performance.

Also, through the individual analysis of the memory and disk cache, it has been possible to make a qualitative analysis, since the speed of the first is significantly higher than the second's, which is a desirable scenario. It is understood that the memory cache attending the user's requests should represent a more responsive and efficient service.

For the network manager, the determination of when the cache loses its efficiency is fundamental, since it can lead to actions to correct this loss of performance as soon as possible. For example, a possible solution for the analyzed scenario is the separation of the student and employee data traffic, since the latter is undetermined by the less predictable behavior of the first. A simple way to achieve this goal without modifying the physical network layout is the use of VLANs, a feature available in the vast majority of layer two switches on the market.

References

[1] Lin YK, Huang CF. Stochastic computer network under accuracy rate constraint from QoS viewpoint. Inf Sci (NY) 2013;239:241–52. doi:10.1016/j.ins.2013.03.033.

[2] Xu L Da, He W, Li S. Internet of Things in Industries: A Survey. IEEE Trans Ind Informatics 2014;10:2233–43. doi:10.1109/TII.2014.2300753.

[3] Zhuming Bi, Li Da Xu, Chengen Wang. Internet of Things for Enterprise Systems of Modern Manufacturing. IEEE Trans Ind Informatics 2014;10:1537–46. doi:10.1109/TII.2014.2300338.

[4] Canali C, Cardellini V, Lancellotti R. Content Adaptation Architectures Based on Squid Proxy Server. World Wide Web 2006;9:63–92. doi:10.1007/s11280-005-4049-9.

[5] Zhu B, Sastry S. Revisit Dynamic ARIMA Based Anomaly Detection. 2011 IEEE Third Int'l Conf. Privacy, Secure. Risk Trust 2011 IEEE Third Int'l Conf. Soc. Comput., IEEE; 2011, p. 1263–8. doi:10.1109/PASSAT/SocialCom.2011.84.

[6] Yaacob AH, Tan IKT, Chien SF, Tan HK. ARIMA Based Network Anomaly Detection. 2010 Second Int. Conf. Commun. Softw. Networks, IEEE; 2010, p. 205–9. doi:10.1109/ICCSN.2010.55.

[7] Pena EHM, Barbon S, Rodrigues JJPC, Proenca ML. Anomaly detection using a digital signature of network segment with adaptive ARIMA model and Paraconsistent Logic. Proc - Int Symp Comput Commun 2014. doi:10.1109/ISCC.2014.6912503.

[8] Fernandes G, Pena EHM, Carvalho LF, Rodrigues JJPC, Proença ML. Statistical, forecasting and metaheuristic techniques for network anomaly detection. Proc. 30th Annu. ACM Symp. Appl. Comput. - SAC '15, New York, New York, USA: ACM Press; 2015, p. 701–7. doi:10.1145/2695664.2695852.

[9] Masuda K, Ishida S, Nishi H. Cross-site Recommendation Application Based on the Viewing Time and Contents of Webpages Captured by a Network Router n.d.

[10] Obaidat MS, Nicopolitidis P, Zarai F. Modeling and Simulation of Computer Networks and Systems, Elsevier; 2015, p. 187–223. doi:10.1016/B978-0-12-800887-4.00034-1.

[11] Fernandez-Prieto J a., Canada-Bago J, Gadeo-Martos M a., Velasco JR. Optimization of control parameters for genetic algorithms to test computer networks under realistic traffic loads. Appl Soft Comput J 2012;12:1875–83. doi:10.1016/j.asoc.2012.04.018.

[12] Zazuli L, Mardedi A. Developing Computer Network Based on EIGRP Performance Comparison and OSPF. Int J Adv Comput Sci Appl 2015;6:80–6. doi:10.14569/IJACSA.2015.060910.

[13] Ben-Porat U, Bremler-Barr A, Levy H. Computer and network performance: Graduating from the "age of Innocence." Comput Networks 2014;66:68–81. doi:10.1016/j.comnet.2014.03.019.

[14] Pimenta AP, Abe JM, de Oliveira CC. An analyzer of computer network logs based on paraconsistent logic. vol. 460. 2015. doi:10.1007/978-3-319-22759-771.

[15] Pimenta Jr AP, Abe JM. Determination of operating parameters and performance analysis of computer networks with Paraconsistent Annotated Evidential Logic Eτ. IFIP Adv Inf Commun Technol 2016;1:1–9.

[16] Misra a. K, Verma M, Sharma A. Capturing the interplay between malware and anti-malware in a computer network. Appl Math Comput 2014;229:340–9. doi:10.1016/j.amc.2013.12.059.

[17] Abe JM, Akama S, Nakamatsu K. Introduction to Annotated Logics - Foundations for Paracomplete and Paraconsistent Reasoning. 1st ed. Springer International Publishing; 2015. doi:10.1007/978-3-319-17912-4.

[18] Abe JM. Paraconsistent Intelligent Based-Systems: New Trends in the Applications of Paraconsistency. Germany: Springer-Verlag; 2015.

[19] Akama S. Towards Paraconsistent Engineering. Springer International Publishing; 2016. doi:10.1007/978-3- 319-40418- 9.