ANALYTICAL MODELING OF RATE-BASED OVERLOAD CONTROL WITH TOKEN BUCKET TRAFFIC SHAPING ON CLIENT SIDE

Konstantin Samouylov
Yulia Gaidamaka
Pavel Abaev
Margarita Talanova
Department of Applied Informatics and Probability Theory
Peoples' Friendship University of Russia
Mikluho-Maklaya str., 6
Moscow 117198, Russia
E-mail: {ksam, ygaidamaka, pabaev, mtalanova}@sci.pfu.edu.ru

Oleg Pavlotsky
Department of communication networks and commutation systems
Moscow Technical University of Communication and Informatics
Aviamotornaya ul., 8a
111024, Moscow, Russia
Emil: oleg.pavlotsky@yandex.ru

KEYWORDS
SIP, overload control, rate-based scheme, Monte-Carlo method, simulation, MMPP.

ABSTRACT

Providing multimedia services to users assumes perfect quality of transfer and modern processing equipment that involves need of development of new methods of the analysis and forecasting of functioning of a communication network. In the paper we investigate the problem of the overload that occurs on a SIP (Session Initiation Protocol) server. SIP overload mitigation is based on RBOC (Rate-based overload control) schema, which was designed by IETF SIP overload control workgroup. The aim of RBOC schema is to restrict signaling message flow rate from the client side towards the server side. Hysteretic overload control algorithm is used to define trigger point for sending control information by the server to the client. Token bucket algorithm is applied to restrict the number of the tokens that the server grants the client. Simulation model for estimation main performance metrics is developed. Traffic with different type of distribution is used for investigation the behavior of the system with tandem servers. The optimization problem for maximization of processor utilization of SIP server on the dependence of threshold values and the number of token is formulated.

INTRODUCTION

The problem of overload control in SIP signaling networks gives rise to many questions which attract researchers from theoretical and practical point of view. Any mechanism that is claimed to settle this problem down demands estimation of control parameters on which its performance is greatly dependent. This research is motivated by the idea that hysteretic control which is successfully deployed and used in SS7 networks may also be applicable and beneficial for overload control in next generation networks.

Note that in hysteretic mechanism parameters that are subject to estimation are those, which define hysteric threshold position.

It is known that overload conditions that arise occasionally in signaling networks lead to severe loss of service quality, which eventually affects network operators and/or service providers.

There are many research papers that deal with analysis of systems with different overload control mechanisms, including hysteretic policy (Abaev and etc. 2012). In this paper system under consideration is SIP proxy server.

At present, two overload control schemes have been proposed by IETF SIP overload control workgroup – one with flow sifting on the sender side (LBOC, Loss-based overload control) documented in (RFC 7339 2014) and one with restricting the flow rate of signaling messages (RBOC, Rate-based overload control) described in (RFC 7415 2015). However, only the basic principles were described in SOCs' documents and methods for calculation of the control parameters were not specified. The control parameters can be determined based on analysis of mathematical models or as the results of simulation modeling.

Different approaches for control parameters estimation of LBOC scheme are proposed in (Abaev and etc. 2012). The same problem for RBOC scheme is still uninvestigated. In this paper, we propose the simulation model for RBOC schema with hysteretic algorithm at the server side, which is used to define trigger point for sending control information.

This paper is organized as follows. We analyze IETF experience for SIP-signaling overload control problem solutions. Then we investigate overload control techniques, which are implemented on the server and

Proceedings 29th European Conference on Modelling and Simulation ©ECMS Valeri M. Mladenov, Petia Georgieva, Grisha Spasov, Galidiya Petrova (Editors)
ISBN: 978-0-9932440-0-1 / ISBN: 978-0-9932440-1-8 (CD)

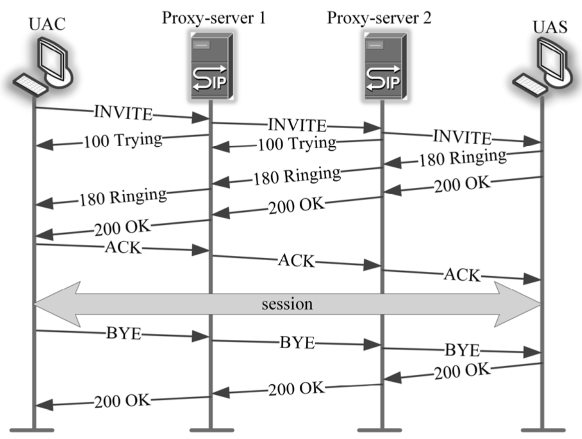
the client side for RBOC scheme. And finally, we introduce the simulation model and numerical example.

RATE-BASED OVERLOAD CONTROL APPROACH OVERVIEW

SIP Call Flow Overview

SIP is a request/response-based protocol. User agents (UAs), which take the role of a user agent client (UAC) or user agent server (UAS) for a request/response pair represent end users. A UAC creates a SIP request and sends it to a UAS. On its way, a SIP request typically traverses one or more SIP proxy servers. The main purpose of a SIP server is to route a request one hop closer to its destination. Responses trace back the path the request has taken.

The UAC sends an INVITE request to the UAS to initiate a SIP session as shown in Fig. 1. Each server on the path confirms the reception of this request by returning a 100 Trying response to the previous hop. Instead of forwarding a request, a SIP server can reject it if it is unwilling or unable to forward the request. Once the request is received by the UAS, it typically responds with a 180 Ringing response to indicate that the called user is being alerted and a 200 OK response when the user has accepted the session. After the 200 OK is received by the UAC, it sends an ACK request to complete the three way handshake of an INVITE transaction. The INVITE request is the only SIP request that uses a three way handshake. Sessions can be terminated at any time by sending a BYE request, which is confirmed with a 200 OK response.



Figures 1: SIP basic call flow

Explicit Overload Control Scheme

The basic idea of RBOC scheme is that receiving entity (RE) informs sending Entity (SE) about the maximum message rate which RE would like to receive from SE within a specified period of time. RE sends the control information to SE periodically depending on RE load changes.

These schemes are based on the idea of feedback control loop between all neighboring SIP servers that directly exchange traffic. Each loop controls only two entities. The Actuator is located on the sending entity and throttles the traffic if necessary. The receiving entity has the Monitor, which measures the current server load.

The four Via header parameters ('oc', 'oc-algo', 'oc-validity' and 'oc-seq') are introduced in (RFC7339 2014) to transfer the control information between two adjacent entities.

Four Via header parameters are defined in (RFC7339 2014) and are summarized below:

- 'oc': Used by clients in SIP requests to indicate support for overload control per (RFC7339 2014) and by servers to indicate the load reduction amount in the loss-based algorithm and the maximum rate, in messages per second, for the rate-based algorithm described here.
- 'oc-algo': Used by clients in SIP requests to advertise supported overload control algorithms and by servers to notify clients of the algorithm in effect. Supported values are "loss" (default) and "rate" (optional).
- 'oc-validity': Used by servers in SIP responses to indicate an interval of time (in milliseconds) that the load reduction should be in effect. A value of 0 is reserved for the server to stop overload control. A non-zero value is required in all other cases.
- 'oc-seq': A sequence number associated with the 'oc' parameter.

The problem of estimation the value of these parameters is beyond of the document.

Threshold Overload Control on the Server Side

We apply hysteretic control algorithm to determine sever overload state. The system during operation changes its state depending on the total number of messages n present in it. Choose arbitrary numbers L and H such that $0 < L < H < R$, where R is the buffer capacity. When the system starts to work it is empty, $n = 0$, and as long as the total number of messages in the system remains below $H-1$, system is considered to be in normal state, $s = 0$.

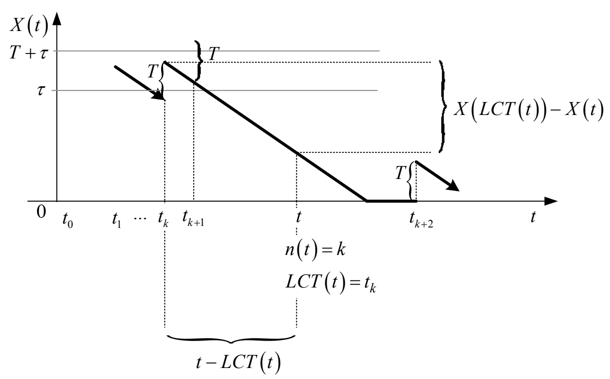
When total number of messages exceeds $H-1$ for the first time, the system changes its state to overload, $s = 1$, and RE informs SE that traffic load should be reduced: it stays in it as long as the number of messages remains between L and $R-1$. Being in overload state, RE's system waits till the number of messages drops down below L after which it changes its state back to normal and informs SE about changes, or exceeds $R-1$ after which it changes its state to blocking, $s=2$, and ask SE for temporary suspension of sending SIP requests. When the total number of messages drops down below $H+1$, system's state changes back to overload, and RE informs SE that the process of sending of messages can be resumed with the current limitations.

Default Algorithm on the Client Side

In the case of RBOC scheme the default Leaky Bucket algorithm (RFC 7415 2015) is proposed to use on the client side to deliver SIP requests at a rate specified in the 'oc' value with tolerance parameter τ. The Leaky Bucket algorithm can be viewed as a finite capacity bucket whose real-valued content drains out at a continuous rate of 1 unit of content per time unit and whose content increases by the increment T for each forwarded SIP request. Parameter T is computed as the inverse of the rate specified in the 'oc' value, namely $T = 1/\text{'oc'}$.

It is assumed that the client tries to put some content into the bucket at random times (time of a message arrival). If at that moment the bucket capacity does not exceed the value τ, the incoming content (incoming message) is added to the bucket and increases the volume of content in the bucket by T. If at that moment the amount of content of the bucket more than τ, the incoming content (message) is not added to the bucket.

The random process of changes of the bucket's content is shown in Fig.2.



Figures 2: Dynamic changes of process $X(t)$

Let $X(t) \geq 0$ is the value of the bucket's content at the moment $t \geq 0$, t_k is the moments of arrival of the *k*-th, $k \geq 1$, message, $LCT(t)$ is the last confirmed time before moment t, $n(t)$ is the number of the latest message accepted before the moment t. These variables satisfy the following relations:

$$n(t) = \sup_{k\geq 1}\{k: \ t_k < t, \ X(t_k) \leq \tau\},$$

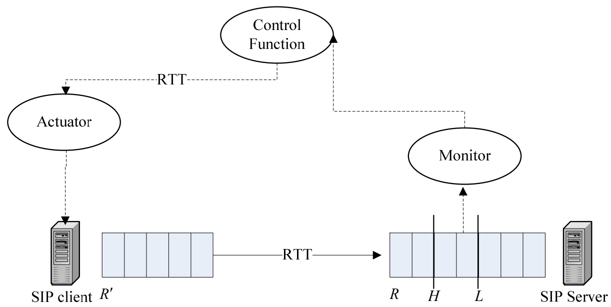
$$LCT(t) = t_{n(t)},$$

$$X(t) = X(LCT(t)) - [t - LCT(t)].$$

Note that the value τ is configured on the client side and chosen depending on the amount of the client's memory and its' processor performance.

SIMULATION MODEL

We model SIP server as a single server queuing system with the finite capacity buffer. Two thresholds L and H are introduced in the buffer. SIP client modeled as a single server with unlimited capacity buffer sends messages to server while free tokens are available. The scheme of the simulation model of messages processing by a SIP-server is shown in Fig. 3.



Figures 3: Simulation model of SIP server with RBOC schema and hysteretic algorithm

All represented in Fig. 3 modules correspond to recommendation (RFC6357 2011), thus the Source of messages and the Actuator belong to SIP client, the Processor with the buffer and the Monitor – to the overloaded downstream SIP server. The Control Function is realized on both SIP-servers, however, in this case, only the downstream SIP server carries out actions by the Control Function, and the Control Function instance on the upstream SIP-server simply passes along feedback. For throttling message flow from the client to the server, we use token bucket algorithm, which shares the same idea as Leaky Bucket. In simulation model, the RTT, which affects the transfer message delay between the client and server, is taken into account. The simulation model operates as follows.

The Source generates messages with the defined distribution of time intervals between adjacent messages. The messages consistently arrive in the Processor buffer. The Monitor measures the queue length in the buffer and when the H buffer occupancy is increasing and crosses the threshold the Monitor detects overload and reports about the situation to the Control Function. The Control Function instance on the server side specifies a number of tokens that will be passed to the client side. In addition, the Control Function also specifies the time interval of token validity. Each token can be used once for granting capability to send message from the client to the server.

Operating time of the simulation model is limited to number T of generated messages.

The simulation model operates to the following rules.

- Input parameters: L, H, R, RTT and traffic profile parameters.
- The client generates a message at the moment t_k+0, the interval $|t_{k+1}-t_k|, k>1$ is exponential or MMPP distributed. The messages are stored in the buffer and sent to the server if token is available.
- The Monitor fixes buffer occupancy $Q(t_k)$ in the system at the moment t_k just before generating a new message.
- The Control Function according hysteretic policy provides some number of free tokens, M, and passes them to the client.
- The transmission delay between the client and server sides is equal to RTT/2.

Two types of traffic generators are implemented by default – exponential and MMPP2 distributions (Fischer 1993). Exponential distribution is described by the mean arrival rate λ^{-1}, and MMPP2 distribution is parametrized by intensity matrix $\begin{pmatrix} 0 & 0 \\ 0 & \lambda \end{pmatrix}$ and infinitesimal operator $\begin{pmatrix} -\alpha & \alpha \\ \beta & -\beta \end{pmatrix}$. Let $\gamma = \dfrac{\beta}{\alpha+\beta}$ denote the burstiness coefficient.

The simulation tool represents the event-driven application [17] realized in C ++ using Microsoft Visual Studio 2012 shell. Each component of the system (SIP processor, Monitor, Control Function, Source and Actuator) is realized as an independent object. An event in this model corresponds to the arriving a message in the buffer or to the changing the system state.

The interface of the simulation tool is presented in Fig. 4.

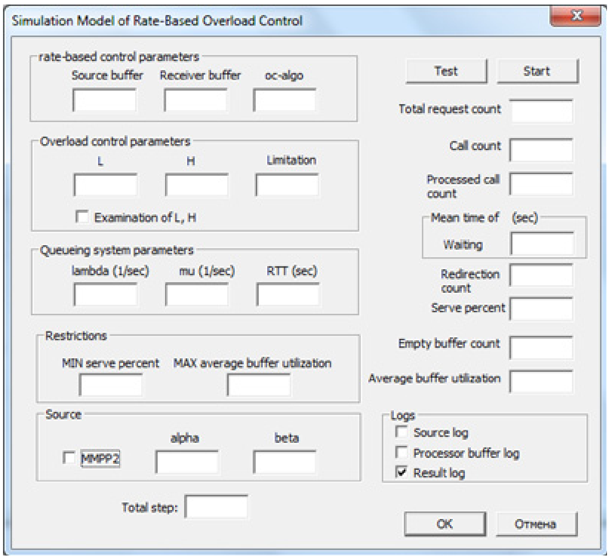


Figure 4: Simulation tool user interface

The buffer occupancy, number of tokens and system state dynamic changes are shown in Fig. 5. In Fig. 6 is shown the dependence of mean waiting time and utilization of the processor on burstiness coefficient γ.

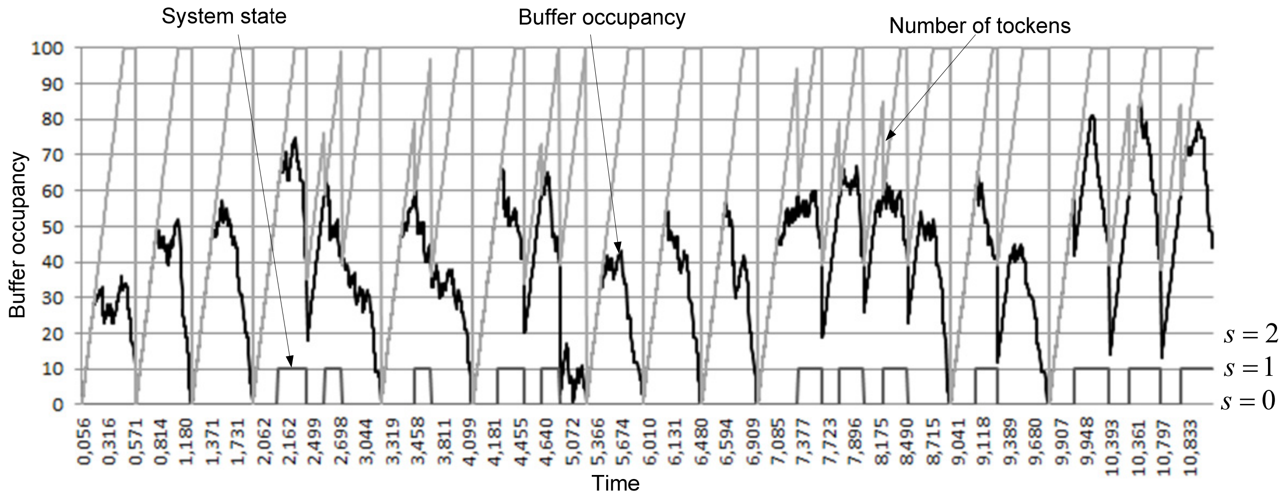


Figure 5: Buffer occupancy dynamic for MMPP2 input flow

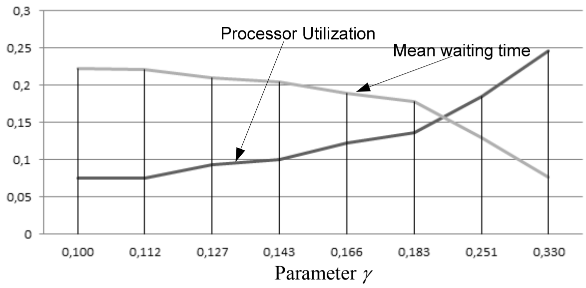


Figure 6: Dependence of mean waiting time on parameter γ

The graph shows that with the increase of variable γ the mean waiting time decreases.

The number of messages processed depends on process utilization. This means that the performance of the server can be evaluated through process utilization. The following optimization problem should be solved to estimate the number of tokens that are passed from the server to the client and the value of the thresholds, L and H :

$$UTIL(L,H,M,\gamma) \to \max$$
$$R1: M_{\min} < M < M_{\max},$$
$$R2: B_{server} < B^*,$$

where B_{server} is the blocking probability.

SUMMARY

In this paper, we give an overview of the SIP protocol and basic 503 mechanisms shortcomings. RBOC simulation model based on Monte-Carlo method was designed. For further study the optimization problem is to be solved and the influence of the burstiness coefficient is to be investigated.

Notes and Comments. This work was supported in part by the Russian Foundation for Basic Research (grant 15-07-03608).

REFERENCES

Abaev, P., Gaidamaka, Yu., Samouylov, K. 2012. "Modeling of Hysteretic Signaling Load Control in Next Generation Net¬works. Lecture Notes". In Computer Science. Germany, Hei¬delberg, Springer-Verlag. -Vol. 7469. 371-378.

Abaev, P., Gaidamaka, Yu., Samouylov, K. 2012. "Queuing Model for Loss-Based Overload Control in a SIP Server Us¬ing a Hysteretic Technique. Lecture Notes". In Computer Sci¬ence. Germany, Heidelberg, Springer-Verlag. -Vol. 7469. 440-452.

Abaev, P., Gaidamaka, Yu., Pechinkin, V., Razumchik, R., Shorgin, S. 2012. "Simulation of overload control in SIP server networks". In Proceedings of the 26th European Conference on Modelling and Simulation, ECMS 2012. Germany, Koblenz. 533-539.

Abdelal, A. and Matragi, W. 2010. "Signal-Based Overload Control for SIP Servers". In Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE. IEEE, Las Vegas, NV, 1-7.

Azhari, S.V. and Nemati, H. 2012. "Stability analysis of tandem SIP proxies". In Communications (ICC), 2012 IEEE International Conference on. IEEE, Ottawa, ON, 1244-1248.

Baghadi, M. and Azhari, S.V. 2013. "Signaling during overload using capabilities of connection-oriented transport protocol". In Electrical Engineering (ICEE), 2013 21st Iranian Conference on. IEEE, Mashhad, 1-6.

Gurbani, V., Hilt, V., Schulzrinne, H. 2013. Session Initiation Protocol (SIP) Overload Control. RFC 7339.

Hilt, V., Noel, E., Shen, C., Abdelal, A. 2011. Design Considerations for Session Initiation Protocol (SIP) Overload Control. RFC 6357.

Hilt V. and Widjaja I. 2008. "Controlling overload in networks of SIP servers". In Network Protocols, 2008. ICNP 2008. IEEE International Conference on. IEEE, Orlando, FL, 83-93.

Montazerolghaem, A. and Yaghmaee M.H. 2013. "SIP overload control testbed: Design, building and Evaluation". In International Journal of Ambient Systems and Applications, Vol. 1, No.2 (June), 17-26.

Noel, E., Williams, P. 2015 Session Initiation Protocol (SIP) Rate Control. RFC 7415.

Ohta, M. 2008. "Performance Comparisons of Transport Protocols for Session Initiation Protocol Signaling." In Telecommunication Networking Workshop on QoS in Multiservice IP Networks, 2008. IT-NEWS 2008. 4th International (Venice, Italy, Feb. 13-15). IEEE, 148-153.

Rosenberg, J., Schulzrinne, H., Camarillo, G. et al. 2002. SIP: Session Initiation Protocol. RFC 3261.

Rosenberg, J. 2008. Requirements for Management of Overload in the Session Initiation Protocol. RFC 5390.

Fischer W., Meier-Hellstern K. S. 1993. The Markov-Modulated Poisson Process (MMPP) cookbook. Performance Evaluation, 18(2). – Pp. 149-171.

AUTHOR BIOGRAPHIES

KONSTANTIN SAMOUYLOV received his Ph.D. from the Moscow State University and a Doctor of Sciences degree from the Moscow Technical University of Communications and Informatics. During 1985-1996 he held several positions at the Faculty of Science of the Peoples' Friendship University of Russia where he became a head of Telecommunication System Department in 1996. Since 2014 he is a head of the Department of Applied Informatics and Probability Theory. His current research interests are probability theory and theory of queuing systems, performance analysis of 4G/5G networks, teletraffic of triple play networks, and signaling networks planning. He is the author of more than 100 scientific and technical papers and three books. His email address is: ksam@sci.pfu.edu.ru.

YULIYA GAIDAMAKA received the Ph.D. in Mathematics from the Peoples' Friendship University of Russia in 2001. Since then, she has been an associate professor in the university's Applied Probability and Informatics department. She is the author of more than 50 scientific and conference papers. Her research interests include SIP signaling, multiservice and P2P networks performance analysis, and OFDMA based networks. Her email address is ygaidamaka@sci.pfu.edu.ru.

PAVEL ABAEV received his Ph.D. in Computer Science from the Peoples' Friendship University of Russia in 2011. He is an Assistant Professor in the Department of Applied Probability and Informatics at Peoples' Friendship University of Russia since 2013. His current research focus is on signaling networks congestion control, performance analysis of wireless 4G/5G networks and M2M communications, applied probability and queuing theory, and mathematical modeling of communication networks. His email address is: pabaev@sci.pfu.edu.ru.

MARGARITA TALANOVA is a PhD student at the Applied Probability and Informatics department, Peoples' Friendship University of Russia. She received the M.Sc. in Computer Science in 2013. Her research interests include performance analysis of SIP networks and congestion control algorithms. Her email address is mtalanova@sci.pfu.edu.ru.

OLEG PAVLOTSKY is a PhD student at the Department of communication networks and commutation systems, Moscow Technical University of Communication and Informatics. He received the M.Sc. in Computer Science in 2013. His research interests include performance analysis of SIP networks and congestion control algorithms. His email address is: oleg.pavlotsky@yandex.ru.