Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (online), 2658-4670 (print)

2022, 30 (3) 244–257
http://journals.rudn.ru/miph

Research article

UDC 519.872:519.217
PACS 07.05.Tp, 02.60.Pn, 02.70.Bf
DOI: 10.22363/2658-4670-2022-30-3-244-257

Development and analysis of models for service migration to the MEC server based on hysteresis approach

Dmitry S. Poluektov, Abdukodir A. Khakimov

Peoples' Friendship University of Russia (RUDN University),
6, Miklukho-Maklaya St., Moscow, 117198, Russian Federation

(received: June 30, 2022; revised: July 18, 2022; accepted: August 8, 2022)

Abstract. Online video services are among the most popular ways of content consumption. Video hosting servers have a very high load every day, which we propose to reduce by migrating the application with the video content in demand to the local Multi-access Edge Computing (MEC) server of the target. This makes it possible to improve the quality of services (QoS) provided to users by reducing the transmission delay. Therefore, an architecture has been proposed that allows, at times of increased demand for the same video content, to migrate the video service application to the edge servers of the network operator. To evaluate the performance of this approach, a mathematical model was developed in the form of a queuing system. The results of the numerical experiment make it possible to optimize the time of using local MEC servers to provide video content.

Key words and phrases: queuing system, service migration, MEC, Markov process, truncated Markov process, video content

1. Introduction

In the modern world, demand for various multimedia services is increasing every year. For example, services for providing online video content are very popular and allow us to access information in a simple way anytime and anywhere from a device with Internet connection. However, with the growth in the amounts of video content and an increase in its demand, requirements for the quality of the services (QoS) are as well increased. Video service providers, in turn, are trying to reduce transmission delays and improve the quality of the video, which increases the size of video files and requires more channel bandwidth to be transmitted to the end user.

The idea of decentralized content placement by a video service provider is not new. In most countries, large cities, operators use the services of geographically distributed content delivery network architecture (CDN). It allows for the video data delivery optimization by using servers, located

much closer to the end user. Multi-access Edge Computing (MEC) servers utilization allows service providers to optimize the transmission process by placing the user-requested content on servers not only within a specific city, but also within a specific district or street. In the same way it solves the problem of high load on transport networks, which is beneficial for the video service provider, who gets the opportunity to provide high-quality content. Thus, the transport network operator can reduce network operation costs and receive additional profit for renting edge computing servers.

MEC introduces the cloud computing capabilities and IT service environment at the edge of the mobile network. The network edge includes base station infrastructure and data centers close to the radio network.

In work [1], the authors have presented a classification of application models and a study of the latest models of mobile cloud applications. In [2], a brief analysis of the requirements for mobile cloud computing (MCC) have been done, the main applications and upload technologies, the classification of contexts and context management methods. In [3], the authors have provided an overview of the definitions, architectures, and applications of MCC, as well as common problems and some existing solutions. In reference [4] a study of existing work on MCC platforms and intelligent access schemes can be found. Another group of scientists in [5] has investigated a detailed taxonomy of mobile cloud computing based on key issues and approaches to address them. Work [6] has introduced a comprehensive overview of the current MCC authentication mechanism and compared cloud computing. The authors in [7] have studied a taxonomy of MEC based on various aspects, including its characteristics, access technologies, applications, purposes, etc. Reference [8] has categorized deployed applications in MEC according to the technical metrics of MEC and the benefit brought by MEC to network stakeholders. A discussion of threats and security in boundary paradigms, as well as possible solution for each specific problem, can be found in [9]. In [10], representative applications and various aspects of the study of fog computing problems are highlighted. An overview on emerging security and privacy issues in fog computing, as well as cloud computing issues is closely discussed in [11]. A study of web caching and prefetching methods for improving network performance, as well as a classification of web caching policies, can be found in [12]. A description of the advantages and disadvantages of cache replacement strategies can be found in [13]. The model of interaction between the edge computing based on Software-defined networking (SDN) and Network functions virtualization (NFV) technology and the cloud computing in the next generation Internet of Things (IoT) is presented in [14]. In [15] authors consider the usage of a MEC server for processing home health monitoring data locally, making it possible to optimize the system-wide cost and the number of patients benefiting from MEC.

2. System description

As described above, the consumption of online video content is growing every year, since any information presented by a video sequence with sound accompaniment makes it easier to perceive or just spend leisure time. Requests for the provision of such services, especially of an entertainment nature, do

not have a constant intensity, but in most cases occur in avalanche bursts at different times or days. For example, in the evening, most people come home from work and watch their favorite series, talk shows, etc. This section describes the process of providing online video and possible scenarios for optimizing content delivery to users.

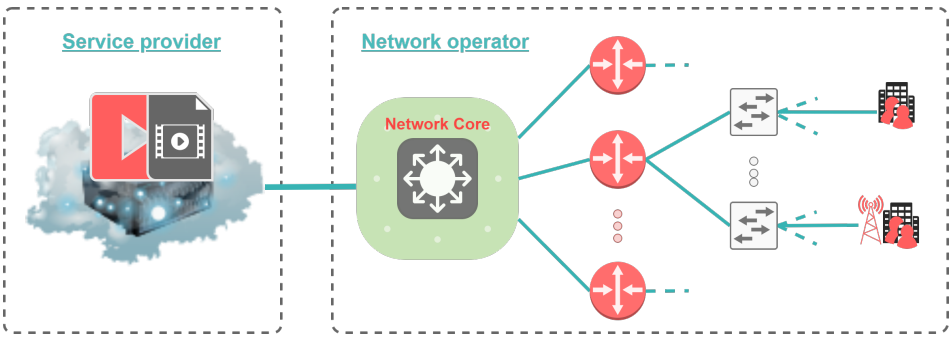

Figure 1. Network architecture for connecting users to a video content service

The figure 1 shows a diagram for providing online video services to users. On the left side the service provider's servers are shown. These servers host and process videos to provide the users on demand. On the right is the operator's last mile access network, which provides end users with access to the global network, and, in particular, connections to the service provider's servers. This network is presented in more detail and consists of

— 1st segment — it includes all elements of the operator's network core and, is responsible for routing traffic within the network and outside of it;
— 2nd segment — it consists of terminal switching devices for wired connection of users and/or base stations of a cellular network for a wireless connection of mobile users (this segment is variable and changes depending on the task).

Between the service provider and the network operator, there are backbone operators and traffic exchange points, which are shown as a direct connection, since they mainly just make data transfer delay, and are also not the main beneficiaries in optimizing the process of providing a service to users.

We consider the process of providing a service to users. The very process of establishing a connection for an online video service has been described and studied in detail in work [16]. Thus it is proposed to focus on the main points presented in the figure 2:

1) a user in the carrier's network sends a video viewing request to a service provider;
2) the service provider processes the request and sends a connection confirmation to the user;
3) a connection is established and the user starts watching the video.

The network operator serves several zones (districts) with N users who can potentially start watching video content. Most of the time, this service is not in great demand, which means that it does not process a large load on the operator's network and service provider's servers (the content delivery method shown in the figure 2). Nevertheless, at some point, a large number

of users start watching the same video content at the same time (for example, the release of a popular series or talk show), thereby a very high load on the servers and the network takes place.

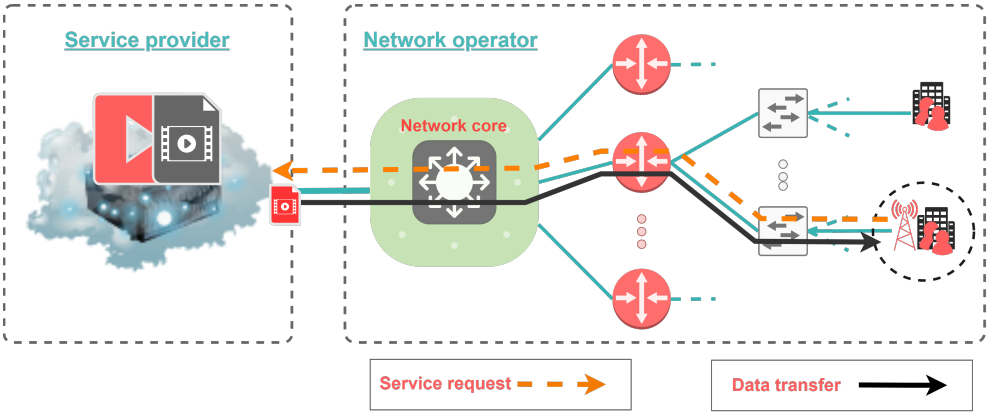

Figure 2. Data transmission for centralized online video providing

Such an avalanche surge of requests leads to an increase in the delay in the transmission of video content and a decrease in bandwidth available for each user, due to the limited bandwidth of the operator's network channel.

To reduce the load on the resources of the service provider and the operator's network, we propose placing MEC servers (figure 3) in each boundary switching zone. This allows us to temporarily migrate the application with access to the video to the facilities of the network operator, located close to the users. Then, the process of content delivery comes down to establishing a connection and transmitting video from MEC server for each high load zone.

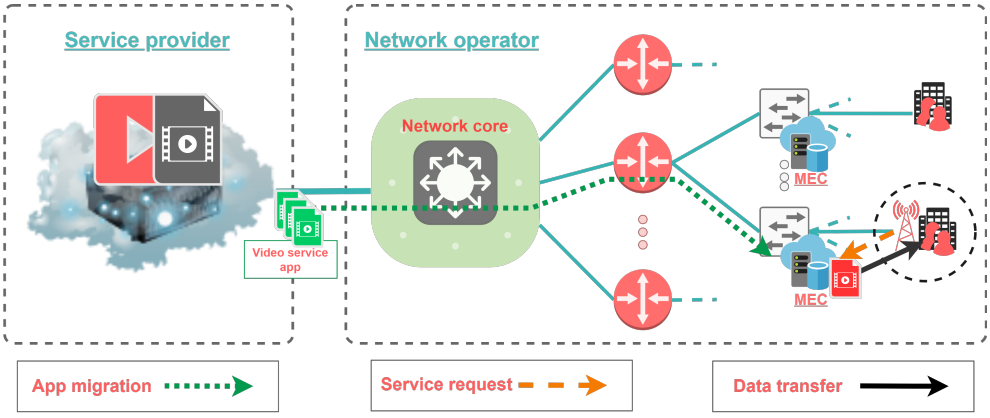

Figure 3. Data transmission for local online video providing with MEC

To formulate a system model, let's consider a single zone of the network operator in which there are N users who can potentially request the same service. $H < N$ users can watch video content directly from the service operator's servers. As soon as $n > H$ users request the service, the video service application is migrated to the MEC server, and all users are already

watching video from the local MEC server. Eventually, the demand for a large number of users in the service disappears, then when the number of active users $n \leqslant L$ (L — the threshold for the appropriate use of the MEC server) is reached, the process of deleting the service from the MEC server and switching the remaining users to the service provider's main server is initiated. The process of deleting and switching does not occur immediately if the number of users becomes $n \leqslant L$ during a certain period of time.

Figure 4 shows the sets of states of the system model and the transitions between them:

— $\mathcal{X}_0$ — serving users directly from the service provider's servers, without using MEC;
— $\mathcal{X}_1(L)$ — disable MEC, switch users to the main server;
— $\mathcal{X}_1(L, H)$ — services from the MEC server without the ability to switch to the main server.

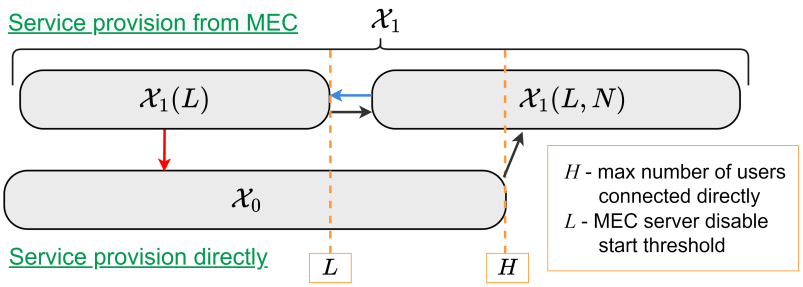

Figure 4. The set of system states for migrating a video service application to MEC

3. Queuing model service migration with hysteresis loop

To analyze the performance of the described system, we model it in the form of a queuing system (QS) model for migration of an application from a remote service provider server to the network operator's MEC server. The system receives a flow of user requests to provide streaming video services. This flow is assumed to be Poisson distributed with mean λ. When serving in the $\mathcal{X}_0$ group and reaching the number of users in the system $n > H$, users are served in the states $\mathcal{X}_1(L, H)$. Video content viewing duration by one user is exponentially distributed with average μ^{-1} minutes. When the system switches to the $\mathcal{X}_1(L)$ state group, the process of disabling the MEC server and switching user services to the service provider's server in the $\mathcal{X}_0$ state group is initialized. It takes an average of α^{-1} minutes to shut down the MEC server correctly, and this parameter is also exponentially distributed.

To analyze the queuing system, we introduce the Markov process $\mathcal{X}(t)$, which describes the behavior of the system at time t, with the state space:

$$\mathcal{X} = \mathcal{X}_0 + \mathcal{X}_1,$$

where

$$\mathcal{X}_0 = \{(s, n) \in \mathcal{X}_0 : \ s = 0, n = (0, H)\},$$

$$\mathcal{X}_1 = \{(s,n) \in (\mathcal{X}_1(L) \cup \mathcal{X}_1(L,H)) : \ s = 1, n = (1,N)\}.$$

The state transition diagram of the Markov process $\mathcal{X}(t)$ is shown in figure 5.

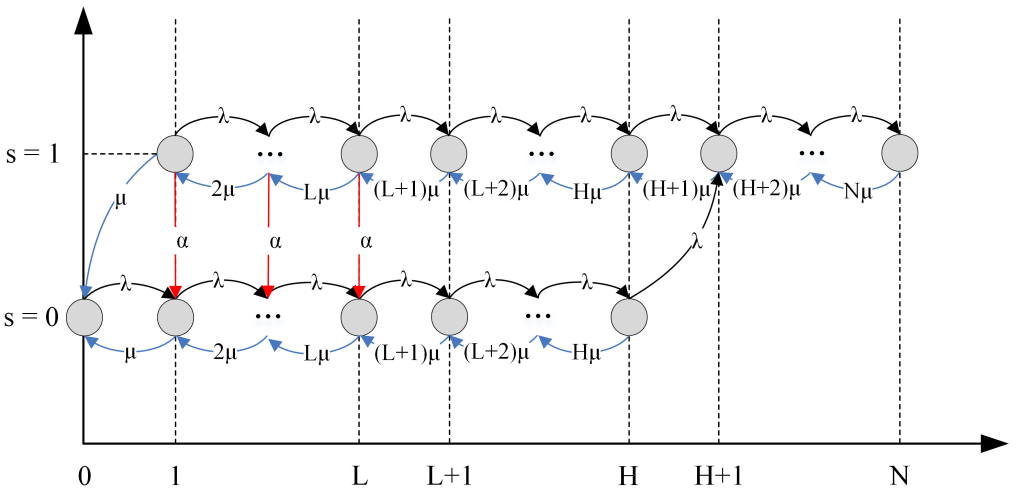

Figure 5. The state transition diagram of the Markov process $\mathcal{X}(t)$

Using the diagram, we write the infinitesimal generator $\mathbf{A}(a_{(s,n),(s',n')} : (s,n)(s',n') \in \mathcal{X})$ of Markov process $\mathcal{X}(t)$. Elements $a_{(s,n),(s',n')}$ are defined as follows:

$$a_{(s,n),(s',n')} = \begin{cases} \lambda, & s' = s,\ n' = n+1 \text{ or } s' = s+1,\ n' = n+1 = H+1, \\ n\mu, & s' = s,\ n' = n-1 \text{ or } s' = s-1,\ n' = n-1 = 0, \\ \alpha, & s' = s-1,\ n' = n \leqslant L, \\ *, & s' = s,\ n' = n, \\ 0, & \text{otherwise}, \end{cases}$$

where $* = -(\lambda \cdot \mathbf{1}\{n < N\} + n\mu + s\alpha \cdot \mathbf{1}\{n \leqslant L\})$.

The system probability distribution $p_{s,n}$ states $(s,n) \in \mathcal{X}$ is numerically calculated using the system of equilibrium equations

$$\begin{cases} \lambda p_{0,0} = \mu p_{0,1} + \mu p_{1,1}, \\ (\lambda + n\mu) p_{0,n} = n\mu p_{0,n+1} \cdot \mathbf{1}_{n<H} + \lambda p_{0,n-1} + \alpha p_{1,n} \cdot \mathbf{1}_{n \leqslant L}, \quad n = (1,H), \\ (\lambda + n\mu + \alpha) p_{1,n} = n\mu p_{1,n+1} \cdot \mathbf{1}_{n<N} + \\ \qquad\qquad + \lambda p_{1,n-1} \cdot \mathbf{1}_{n>1} + \lambda p_{0,H} \cdot \mathbf{1}_{n=H+1}, \quad n = (1,N), \\ \sum\limits_{(s,n) \in X} p(s,n) = 1. \end{cases}$$

An important performance metric of the considered system is the time spent by users in the set of $\mathcal{X}_1$ states, which in the system model corresponds to the time that users watch video content from the MEC server until it is turned off. That is the lifetime of the service provider application after its migration to MEC. In the mathematical model, this is equal to the time interval from the moment when the Markov process $\mathcal{X}(t)$ reached the number of customers in the system H and passed into the set $\mathcal{X}_1$, i.e. into the state $(1, H+1)$, until the moment when the process returned back to the set $\mathcal{X}_0$.

Let us denote τ_1 a random variable of the sojourn time of requests in the set $\mathcal{X}_1$. In order to find the cumulative distribution function (CDF) F_{τ_1} of the random variable τ_1, we can describe our systems by a truncated Markov process $\hat{\mathcal{X}}(t)$, which describes the behavior of the system at time $t > 0$ with the state space:

$$\hat{\mathcal{X}} = \mathcal{X}_0 + \hat{\mathcal{X}}_1^B,$$

where $\hat{\mathcal{X}}_1^B = \{(0, n) : n = 1, \dots, L\}$.

The state transition diagram of the truncated Markov process $\hat{\mathcal{X}}(t)$ is shown in figure 6.

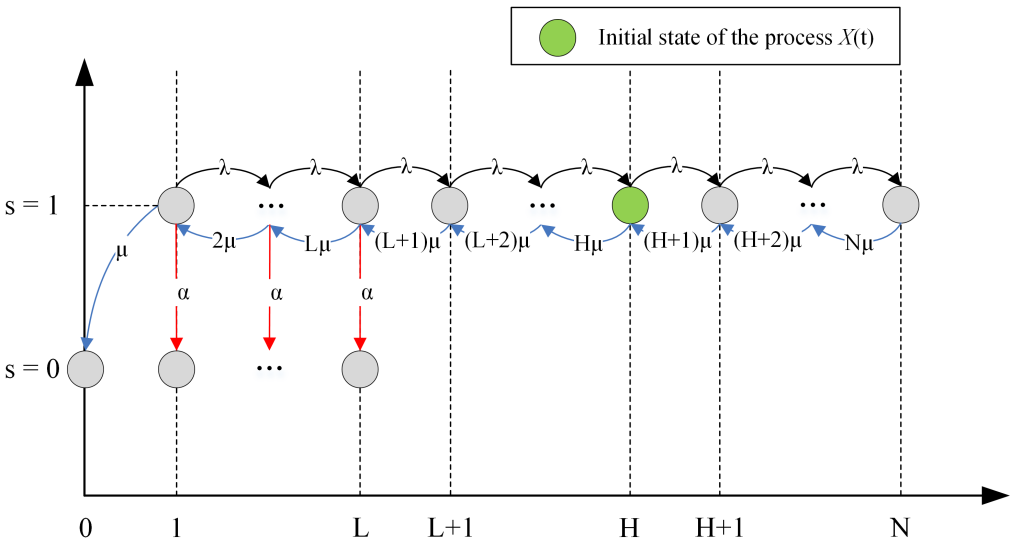

Figure 6. The state transition diagram of the truncated Markov process $\hat{\mathcal{X}}(t)$

The matrix $\hat{\mathbf{P}}(t)$ of transition probabilities can be written as follows:

$$\hat{\mathbf{P}}(t) = e^{\hat{\mathbf{A}}} = \sum_{n=0}^{\infty} \frac{\hat{\mathbf{A}} t^n}{n!}, \quad t \geqslant 0, \tag{1}$$

where $\hat{\mathbf{A}}$ is the infinitesimal operator of process $\hat{\mathcal{X}}(t)$. The distribution $\hat{\mathbf{p}}(t)$ of truncated process $\hat{\mathcal{X}}(t)$ satisfies the following equations:

$$\hat{\mathbf{p}}^T(t) = \hat{\mathbf{p}}^T(0)\hat{P}(t), \tag{2}$$

$$\frac{d}{dt}\hat{\mathbf{p}}^T(t) = \hat{\mathbf{p}}^T(0)\hat{\mathbf{A}}e^{\hat{\mathbf{A}}t}, \quad t \geqslant 0. \tag{3}$$

Initial probability vector $\hat{\mathbf{p}}^T(0)$:

$$\hat{\mathbf{p}}_{(s,n)}(0) = \begin{cases} 1, & (s,n) = (1, H+1), \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Afterwards, we can find the cumulative distribution function $F_{\tau_1}(t)$ of the random variable τ_1 which equals

$$F_{\tau_1}(t) = \sum_{i=1}^{L} p_{(s,n)}(t), \quad t \geqslant 0. \tag{5}$$

The probability density function (PDF) of the random variable τ_1 is given by:

$$f_{\tau_1}(t) = \mu p_{(0,0)}(t) + \lambda \sum_{i=1}^{L} p_{(0,i)}(t). \tag{6}$$

We calculate the average time before the MEC shutdowns through the expectation of the random variable τ_1:

$$W_{\tau_1} = E(\tau_1) = \int_0^\infty t f_{\tau_1}(t)dt = \mu \int_0^\infty t p_{(0,0)}(t)dt + \lambda \sum_{i=1}^{L} \int_0^\infty t p_{(0,i)}(t)dt. \tag{7}$$

4. Numerical analysis

For numerical analysis, consider the network operator's service area, which is home to N users who are fans of the same series. New episodes of the series are released once a week and most users try to watch it as soon as it's possible, thereby creating a high load on the service provider's servers. In our scenario, every λ^{-1} minute there is a request to watch a video content. The duration of watching a video depends on the length and fascination of the episode, which takes μ^{-1} minute on average. The allowable load on the service provider's servers equals to H of user requests, when the limit is reached, the video content application is migrated to the nearest MEC server of the network operator. At some point, many users stop watching the series and the number of active users decreases. As soon as their number reaches the L threshold of active sessions, it becomes unreasonable to provide the service through the local MEC server and the disconnection process begins, which takes α^{-1} minutes on average.

For a more accurate estimate of the system, we introduce the parameter $\rho = \lambda/\mu$ describing the ratio of the rate of requests to watch a video to its average duration. This way, it is possible to determine the load created by users, which in our case correlates with the average number of active sessions.

We consider how the video viewing time through MEC changes at different values of the threshold for the start of its shutdown. The initial data is presented in the table 1.

Table 1

Initial data

Notation	Value	Description
N	300	Maximum number of users watching videos
H	100	The max number of active user sessions on the service provider's server
L	85–100	MEC server disable threshold
μ^{-1}	45	Average video watch time, min
ρ	95, 100, 105, 110, 115	System load, number of active sessions
α^{-1}	15	average MEC disabling time, min

Figure 7 shows a plot of the average user service time through the MEC server depending on the threshold L value for the start of its disconnection at different loads (number of active sessions). It can be seen that as L increases, the MEC usage time decreases non-linearly, and the closer to the threshold H, the less significant the change. Periods of high load at which it is necessary to use MEC usually occur in the evening hours, therefore, it is proposed to choose the optimal time to use the MEC server in the range from 120 minutes (2 hours) to 300 minutes (5 hours), which is marked with a dotted line. With a value of 110 and 115 average number of active sessions, the disconnect threshold required to fall within the specified range is 97–100 users, which is quite close to threshold H. This shows the effectiveness of using MEC under such a load, but imposes increased costs on the service provider.

It is also important to evaluate the impact of the average duration of video viewing and the duration of the MEC server shutdown on the time the service is provided through MEC. To do this, consider an optimally loaded system with a load $\rho = 100$ and several threshold values $L = 85, 90, 95, 100$, at which the average service time through MEC falls within the interval in figure 7.

The figure 8 depicts MEC server runtime under different values of threshold L. The average video viewing time is between 45 minutes and 2 hours, which corresponds to typical episodes of a TV series or a full-length film. It can be seen that while maintaining the average number of active sessions, an increase in the duration of each session can significantly affect the time of using MEC, especially at sufficiently low values of the threshold L. This is due to the fact that the total audience capture time becomes longer, and the possibility of disconnecting several users is lower. Although when using $L = H$, the average MEC usage time does not change significantly, which indicates frequent service switching between the service provider's servers and the local MEC servers.

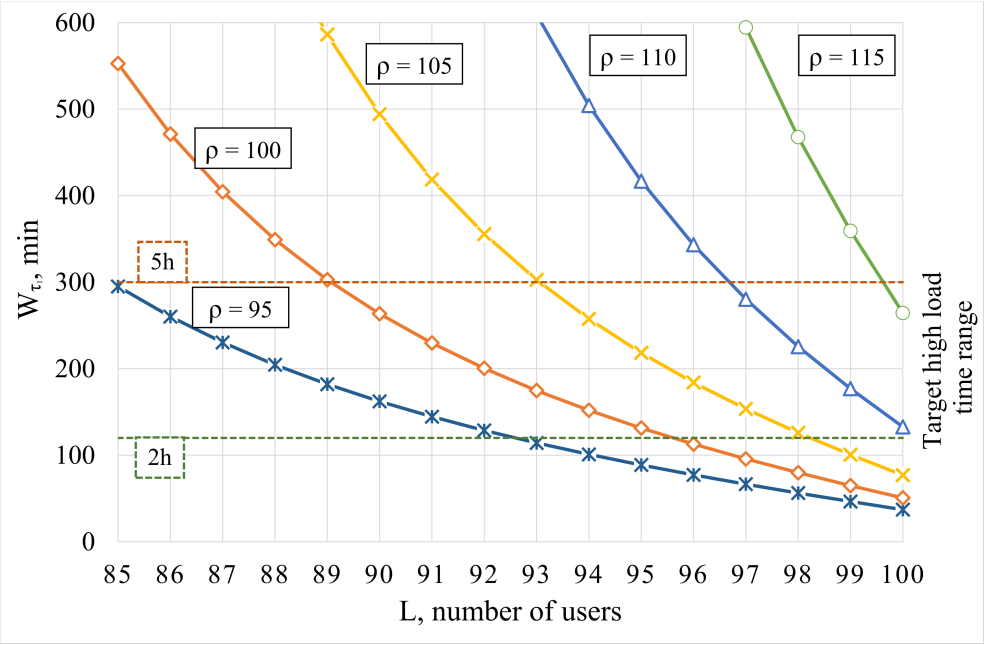

Figure 7. MEC server runtime against the threshold disable value for different ρ

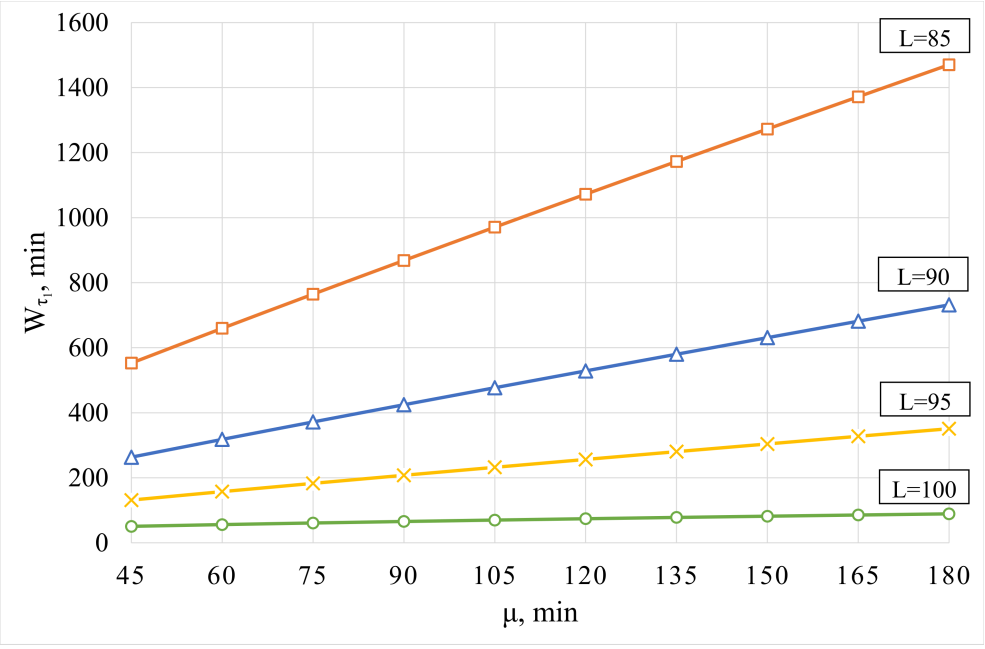

Figure 8. MEC server runtime against the average watch time for different L

Therefore, it is worth taking an assessment of the impact of the MEC disable delay on the duration of its use. In figure 9 this dependence is shown, with similar values of ρ and L for figure 8, average viewing time is 45 minutes

and average time required to disable MEC varies between 5–30 minutes. A similar behavior of the curves can be observed, which shows the expected increase in MEC usage time. It also allows, depending on the L threshold used, to select a more optimal time required to switch service back to the service provider's server.

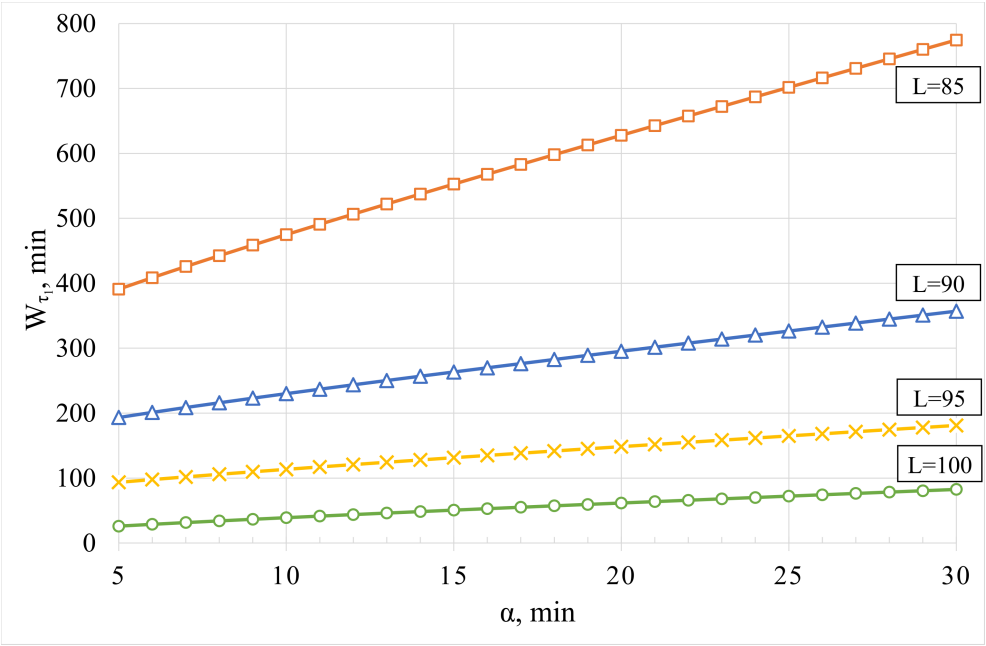

Figure 9. MEC server runtime against the MEC server disabling time for different L

5. Conclusions

The presented paper has investigated the scenario of providing services by a service provider of video content to users using the local MEC servers of the telecommunication operator under high service load. A mathematical model of the interaction for the case described in the scenario in the form of a Markov queuing system with hysteresis control using the MEC server has been developed. The resulting equation was derived for calculating the average time the MEC server is used to provide a video content depending on time. A numerical analysis of the scenario was carried out for one highly loaded zone of the telecommunication operator, in which users massively request to watch a video on an example of a popular TV series. It is shown how the changes in the MEC server disable initialization threshold, the duration of a video viewing and the duration of the MEC disabling have an effect on the average time of using local edge servers. This allows to set the optimization problem for various scenarios in the future.

Also, this work can be considered a continuation of [17], which allows, by placing the MEC server on the UAV, not only to reduce the load on the servers of the video content provider, but also to increase the QoS and QoE parameters for the mobile users.

Acknowledgments

The reported study was funded by RFBR, project number 20-37-90131 (recipient Dmitry Poluektov).

References

[1] A. Khan, M. Othman, S. A. Madani, and S. U. Khan, "A Survey of Mobile Cloud Computing Application Models," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 393–413, 2014. DOI: 10.1109/SURV.2013.062613.00160.

[2] L. Guan, X. Ke, M. Song, and J. Song, "A Survey of Research on Mobile Cloud Computing," in *2011 10th IEEE/ACIS International Conference on Computer and Information Science*, 2011, pp. 387–392. DOI: 10.1109/ICIS.2011.67.

[3] H. Dinh Thai, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, vol. 13, Dec. 2013. DOI: 10.1002/wcm.1203.

[4] X. Fan, J. Cao, and H. Mao, *A survey of mobile cloud computing*, English, 2011.

[5] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile cloud computing: A survey," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 84–106, 2013, Including Special section: AIRCC-NetCoM 2009 and Special section: Clouds and Service-Oriented Architectures. DOI: 10.1016/j.future.2012.05.023.

[6] M. Alizadeh, S. Abolfazli, M. Zamani, S. Baharun, and K. Sakurai, "Authentication in mobile cloud computing: A survey," *Journal of Network and Computer Applications*, vol. 61, pp. 59–80, 2016. DOI: 10.1016/j.jnca.2015.10.005.

[7] A. Ahmed and E. Ahmed, "A survey on mobile edge computing," in *10th International Conference on Intelligent Systems and Control (ISCO)*, 2016, pp. 1–8. DOI: 10.1109/ISCO.2016.7727082.

[8] M. Beck, M. Werner, S. Feld, and T. Schimper, "Mobile Edge Computing: A Taxonomy," in *The Sixth International Conference on Advances in Future Internet*, 2014, pp. 48–54.

[9] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems*, vol. 78, part 2, pp. 680–698, 2018. DOI: 10.1016/j.future.2016.11.009.

[10] S. Yi, C. Li, and Q. Li, "A Survey of Fog Computing: Concepts, Applications and Issues," in *Proceedings of the 2015 Workshop on Mobile Big Data*, ser. Mobidata '15, Hangzhou, China: Association for Computing Machinery, 2015, pp. 37–42. DOI: 10.1145/2757384.2757397.

[11] S. Yi, Z. Qin, and Q. Li, "Security and Privacy Issues of Fog Computing: A Survey," Aug. 2015, pp. 685–695. DOI: 10.1007/978-3-319-21837-3_67.

[12] W. Ali, S. M. Shamsuddin, and A. S. Ismail, "A Survey of Web Caching and Prefetching A Survey of Web Caching and Prefetching," *International Journal of Advances in Soft Computing and its Applications*, vol. 3, no. 1, 2011.

[13] S. Podlipnig and L. Böszörmenyi, "A Survey of Web Cache Replacement Strategies," *ACM Comput. Surv.*, vol. 35, no. 4, pp. 374–398, Dec. 2003. DOI: 10.1145/954339.954341.

[14] Z. Lv and W. Xiu, "Interaction of Edge-Cloud Computing Based on SDN and NFV for Next Generation IoT," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5706–5712, 2020. DOI: 10.1109/JIOT.2019.2942719.

[15] Z. Ning, P. Dong, X. Wang, X. Hu, L. Guo, B. Hu, Y. Guo, T. Qiu, and R. Y. K. Kwok, "Mobile Edge Computing Enabled 5G Health Monitoring for Internet of Medical Things: A Decentralized Game Theoretic Approach," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 2, pp. 463–478, 2021. DOI: 10.1109/JSAC.2020.3020645.

[16] T. Stockhammer, "Dynamic Adaptive Streaming over HTTP: Standards and Design Principles," in *Proceedings of the Second Annual ACM Conference on Multimedia Systems*, ser. MMSys '11, San Jose, CA, USA: Association for Computing Machinery, 2011, pp. 133–144. DOI: 10.1145/1943552.1943572.

[17] A. Khakimov, E. Mokrov, D. Poluektov, K. Samouylov, and A. Koucheryavy, "Evaluating the Quality of Experience Performance Metric for UAV-Based Networks," *Sensors*, vol. 21, no. 17, 2021. DOI: 10.3390/s21175689.

For citation:

D. S. Poluektov, A. A. Khakimov, Development and analysis of models for service migration to the MEC server based on hysteresis approach, Discrete and Continuous Models and Applied Computational Science 30 (3) (2022) 244–257. DOI: 10.22363/2658-4670-2022-30-3-244-257.

Information about the authors:

Poluektov, Dmitry S. — postgraduate student of Department of Applied Probability and Informatics of Peoples' Friendship University of Russia (RUDN University) (e-mail: poluektov-ds@rudn.ru, phone: +7(495)9522823, ORCID: https://orcid.org/0000-0002-4246-8483)

Khakimov, Abdukodir A. — Researcher of Department of Applied Probability and Informatics of Peoples' Friendship University of Russia (RUDN University) (e-mail: khakimov-aa@rudn.ru, phone: +7(495)9522823, ORCID: https://orcid.org/0000-0003-2362-3270)

УДК 519.872:519.217
PACS 07.05.Tp, 02.60.Pn, 02.70.Bf
DOI: 10.22363/2658-4670-2022-30-3-244-257

Разработка и анализ моделей гистерезисного управления миграцией сервисов на сервер граничных вычислений

Д. С. Полуэктов, А. А. Хакимов

Российский университет дружбы народов,
ул. Миклухо-Маклая, д. 6, Москва, 117198, Россия

Аннотация. Сервисы онлайн-видео являются одними из самых популярных способов потребления контента. На серверы видео хостинга приходится ежедневно колоссальная нагрузка, которую нами предложено снизить за счёт миграции приложения востребованным видеоконтентом на локальный сервер MEC целевой зоны. Это позволит повысить качество предоставляемых услуг пользователям за счёт сокращения задержки на передачу. Поэтому предложена архитектура, дающая возможность в моменты повышенного спроса на одинаковый видеоконтент производить миграцию приложения видеосервиса на граничные серверы оператора сети. Для оценки показателей эффективности такого подхода была построена математическая модель в виде системы массового обслуживания. Результаты численного эксперимента позволяют произвести оптимизацию времени использования локальных серверов MEC для предоставления видеоконтента.

Ключевые слова: миграция сервисов, граничные вычисления, марковский процесс, MEC, онлайн-видео, усечённый марковский процесс