Digital Communications and Networks 6 (2020) 189–194

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

Recent advances in mobile edge computing and content caching

Sunitha Safavat, Naveen Naik Sapavath, Danda B. Rawat *

Data Science and Cybersecurity Center (DSC²), Department of Electrical Engineering and Computer Science, Howard University, Washington, DC, 20059, USA

ARTICLE INFO

Keywords:
Mobile edge computing
Content caching
MEC

ABSTRACT

The demand for digital media services is increasing as the number of wireless subscriptions is growing exponentially. In order to meet this growing need, mobile wireless networks have been advanced at a tremendous pace over recent days. However, the centralized architecture of existing mobile networks, with limited capacity and range of bandwidth of the radio access network and low bandwidth back-haul network, can not handle the exponentially increasing mobile traffic. Recently, we have seen the growth of new mechanisms of data caching and delivery methods through intermediate caching servers. In this paper, we present a survey on recent advances in mobile edge computing and content caching, including caching insertion and expulsion policies, the behavior of the caching system, and caching optimization based on wireless networks. Some of the important open challenges in mobile edge computing with content caching are identified and discussed. We have also compared edge, fog and cloud computing in terms of delay. Readers of this paper will get a thorough understanding of recent advances in mobile edge computing and content caching in mobile wireless networks.

1. Introduction

Mobile Edge Computing (MEC) is a decentralized computing concept in which computing resources and application services can be distributed along the communication path from the point storing data to Base Stations (BSs) in wireless networks. This fulfills the computational needs of the end users at the edge when delay-sensitive and low-computational operations are required. MEC allows wireless subscribers to access the closest computing servers within the range of wireless Radio Access Network (RAN) [1]. (see Table 1)

The main goal of MEC is to reduce latency by bringing the computation and storage capabilities of the core network to the edge network. MEC could offer real-time information to different applications, such as real-time network load and real-time location information of wireless users, by using the available edge close to the end users. This real-time network information can be leveraged to offer context-aware applications and services to the mobile network users by meeting Quality of Experience (QoE) of the end users. The MEC platform offers great benefit by reducing latency and back-haul bandwidth consumption, but increases the responsibility of the edge to offer several services to the end users in a real-time manner [2,3]. Wireless network operators can allow edge computing to be handled by third-parties, and this allows service providers and edge services to rapidly deploy new applications for the mobile users. In other words, MEC tries to avoid the use of a centralized data warehouse but allows processing data near the end users (i.e., edge of the network) where the actual data is generated. MEC enables data flow acceleration, including real-time data processing, with low latency. MEC also allows different applications and smart mobile devices to respond, process data and make informed decisions in a near real-time manner as soon as data is generated/created so as to eliminate the lag/delay. This is one of the essential assets for many emerging technologies, such as self-driving cars and real-time navigation systems [4]. MEC reduces the use of back-haul/Internet bandwidth significantly since it can handle large amounts of data near the source. MEC helps to reduce costs and ensures that the applications can operate without accessing costly and high delay back-haul links. Furthermore, MEC offers local storage with the ability to process data without putting it in a remote public cloud. This feature adds an extra layer of security that is useful for sensitive and private data.

Furthermore, content caching within MEC has demonstrated to be beneficial [5]. Due to the exploitation of different types of BSs for MEC, the future edge networks are considered to be diverse. Thus, in edge networks, the caching will be deployed at various places in different BSs. The content requested by users is retrieved from the central server at the beginning if the content is not available at the edge. Then caching is enforced to keep of copy of the content for future use. If the data is accessed from a centralized server every time, a slow back-haul link will introduce significant delay while delivering the content to end users.

* Corresponding author.
E-mail addresses: Sunitha.Safavat@bison.howard.edu (S. Safavat), naveennaik.sapavath@bison.howard.edu (N.N. Sapavath), db.rawat@ieee.org (D.B. Rawat).

https://doi.org/10.1016/j.dcan.2019.08.004
Received 20 January 2019; Received in revised form 2 May 2019; Accepted 29 August 2019
Available online 4 September 2019

Table 1
Summery of the state-of-the-arts for computation offloading and edge caching.

Approach	Key Points
Single User [12]	Computation Offloading Choice
Multi-User [13]	Multi-user computation offloading is NP hard problem
Offloaded to Devices [15]	Device to Device technology
Mobility Awareness [16]	Mobility aware offloading strategy
Caching policies [17]	User's preference based polices
Content popularity [18]	Power law distribution
User preference-based policies [17]	User's preference toward specific video categories

Additionally, with the development and deployment of cheaper storage units and several BSs, deploying caching mechanisms at Small Base Stations (SBSs) as well as at Macro Base Stations (MBSs) has become easy and cost-effective recently. In the emerging wireless networks, device-to-device communications with device-level storage units will allow the user-level content caching to base itself upon interests and some social relations between users [6]. The basic architecture of content caching and edge computing is shown in Fig. 1.

As in edge computing, the content caching strategy to cache the content on the way from its centralized location to end users in cache-enabled mobile networks is crucial to reduce the delay and enhance the QoE for the users. In such cache-enabled mobile networks, caching insertion methods can make a decision whether or not to cooperate with different caching edges and how the content is cached in the edge servers near the users, and caching expulsion strategies determine the dynamics and metrics for the contents that have already been cached. The content selection process considers which contents should be cached, what contents are to be updated, and how long the content should be cached. One of the criteria is content popularity, which is commonly used as a

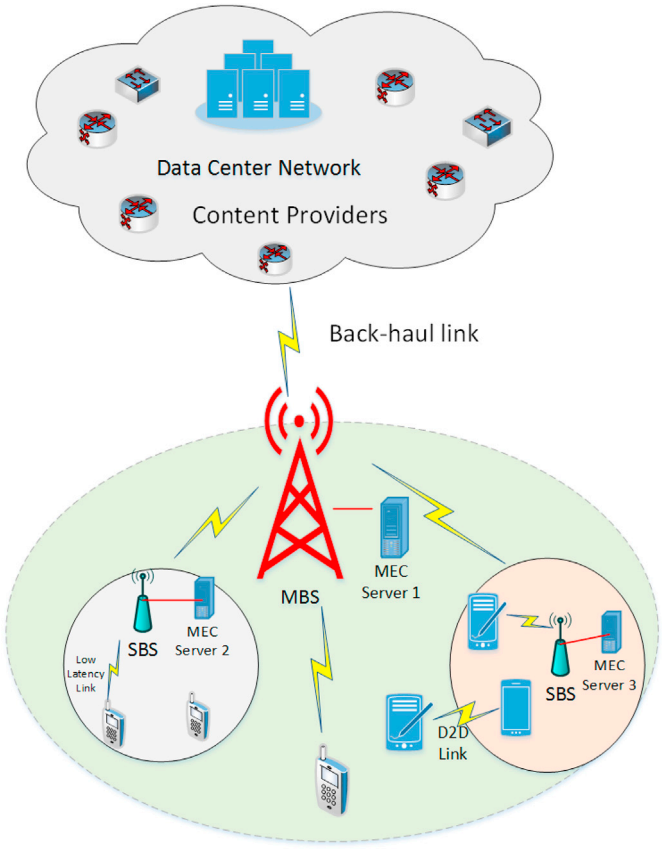

Fig. 1. A simplified Mobile Edge Commuting (MEC) architecture of mobile network with edge computing and content caching.

very important factor for fast content retrieval. Another is content diversity which helps to increase the categories of contents cached regionally.

The network resources, such as cache storage size, computing, energy and communication bandwidth within the MEC-enabled network, should be controlled for their effective usage. For better efficiency, it is required to significantly optimize what data to be cached, how to insert the caching data, and how to evict contents from the cache storage by taking account of data quality, diversity and end user mobility. Caching optimization deals with the problems associated with optimization on networks as well as end user performances, such as network architecture, analytical approaches and content caching strategies.

Specifically, this paper presents a survey on recent advances in mobile edge computing and content caching in mobile wireless networks. Some important open challenges in mobile edge computing with caching are identified and discussed. A comparison of edge, fog and cloud computing in terms of delay is presented. With this paper, readers have a thorough understanding of recent advances in edge computing and caching in wireless networks.

The rest of the paper is organized as follows. Computing at the edge network is presented in Section 2. Section 3 presents caching replacement strategies. Section 4 presents the behavior and performance of the caching system. Optimization based on caching networks is presented in Section 4.2. Some comparisons are presented in Section 5, followed by some research challenges in Section 6. Section 7 concludes the paper.

2. Computing in mobile edge networks

This section presents the objectives and computation of offloading techniques in mobile edge networks.

2.1. MEC objectives

The primary goal of MEC is to lower the latency and enhance the QoE for the end users. Different wireless applications or systems may have different performance requirements, such as delay requirements, which can be met through MEC networks. We present performance objectives that edge computing provides in mobile wireless networks in the following subsections.

2.1.1. Minimization of latency

Latency is one of the primary performance measures, which affects the experience of end users [7]. The latency requirement of the 5G wireless network system for a Round Trip Time (RTT) is 1 ms, which is nearly ten times lower than that of 4G. When the content is coming from a centralized server located in the centralized data center to the end users, it takes a longer time compared with that of the MEC-server. Similarly, the delay due to offloading tasks to the cloud is high for the data offloading applications, and the long delay is unacceptable for many applications. In order to minimize the delay/latency, implementing high-density SBSs with edge computing is a more viable approach [7].

2.1.2. Maximization of network capacity

The 5G wireless network is expected to support a thousand times higher volume of mobile data per area than the current 4G network [8]. In order to handle this expected huge data, future wireless networks require higher capacity in the RAN, back-haul and front-haul. Data offloading as well as context-aware computation offloading are a combination of technologies that are expected to address some of the challenges in the RAN on top of utilizing more spectrum with higher spectrum efficiency [9]. MEC and content caching could help increase the network capacity by caching popular content to the edge and BSs, and by saving the back-haul bandwidth [8].

2.1.3. Minimization of energy consumption

Many works have been done to evaluate the energy efficiency of edge

computing (e.g., [10]). Various optimization schemes have been proposed to minimize energy consumption in both networks and individual devices. For computation offloading in the next generation heterogeneous networks, the energy cost associated with task computing and file transmission is regarded as one of the central cost components [10]. It is important to design an energy-efficient data/computation offloading scheme, which jointly optimizes radio resources and energy consumption while minimizing the overall latency. In [10], end devices are classified into three types according to their abilities and requirements. Wireless channels of MBSs and SBSs are allocated to mobile devices according to their priority until all devices receive required channels. At every iteration, the scheme ensures the system obtains minimum energy cost. The results have shown that the proposed scheme has lower energy consumption, particularly with a large number of end users.

2.2. Computation offloading

End devices are typically constrained by limited computing, battery life and storage capacity. One of the main purposes of edge computing is computation offloading to overcome the limitation of mobile devices, such as computational capabilities, battery resources and storage availability [11].

2.2.1. Single user offloading to edge

Scheduling strategies for offloading in the single-user case should be deployed to minimize the delay and energy consumption to combat stochastic channel conditions in wireless networks. In [12], a threshold -based scheduling policy has been proposed to minimize the energy consumption for single- and multi-server scenarios.

2.2.2. Multi-user offloading to edge

The data or computation offloading the multi-user scenario is much more complex and must deal with complex issues from scheduling to allocation compared with a single user offloading scenario [13]. It is known that the multi-user offloading is an NP-hard problem [13]. This problem can be solved using theoretic approaches where a socially optimal equilibrium can be achieved [13].

2.2.3. Offloaded to another device

A device can offload its contents or computation to another nearby device using device-to-device communication to leverage additional computation resources of another device when it is possible. The end devices collected as a group can be used to provide such services, instead of the edge server, the computational tasks could be offloaded to other nearby mobile devices.The scheduling problem for offloading to another device is expected to be different from that of offloading to the server [14].

2.2.4. Offloaded to edge server

Typically offloading of computational tasks is done to edge servers. While selecting the edge servers to minimize delay and energy consumption to maximize QoE for task offloading, we need to consider different parameters, such as CPU cycles, offloading link capacity, energy consumption, cache size, etc. [15].

2.2.5. Mobility awareness in offloading

User mobility is one of the most important features to be considered in the edge network since the mobility determines the connection setup time and dwelling time between users and servers. The mobility of end users results in a dynamically changing network topology, which directly impacts task offloading strategies [16].

3. Caching locations and cache replacement strategies

This section explains various caching placement and replacement policies at different places in MEC-networks.

3.1. Caching locations

In MEC, we could deploy edge servers and content caching within the mobile networks. In typical wireless cellular networks for caching, we could cache the content at the core network, RAN and end-devices [19,20]. With MEC caching and edge server, the data traffic could be reduced significantly [19]. The different locations for caching in MEC networks are discussed in the following subsections.

3.1.1. Micro Base Stations (MBSs)

MBSs in heterogeneous networks are the places where caching and edge server are deployed [17] and where caching can be done reactively and pro-actively. By leveraging the MBS-based caching and edge computing, system capacity could be significantly enhanced, and the delay can be significantly reduced as the content will be available at the edge near the users [17,21].

3.1.2. Small Base Stations (SBSs)

SBSs are expected to be heavily deployed in the next-generation of heterogeneous wireless networks. Caching in SBSs is also a good idea since they are closer to end users, which could serve the users faster with high data rates [22].

3.1.3. End device caching

Device to Device (D2D) communication is also expected in 5G wireless networks. In the D2D framework, end devices could leverage their storage for caching the content, which will significantly reduce delay [23]. Caching in D2D can be done cooperatively [24] among end users by forming clusters or individually.

3.2. Caching insertion strategy

The following subsections present some cache insertion strategies in MEC-enabled systems.

3.2.1. Caching everywhere

Caching everywhere is the least conservative way of caching the contents in the MEC system. It could be considered as a default option for the system where no optimization is needed, from the source to the end user, wherever possible content will be cached, which could introduce extra burden on storage or extra burden on handling the recent caches efficiently [25].

3.2.2. Caching with probability

Most issues on caching everywhere could be handled by using caching with probability so as to enhance the storage efficiency and reduce the caching redundancy. The caching with probability information is utilized and can improve cache efficiency. The content of higher usage probability is cached, while that of lower usage probability is not cached [26].

3.2.3. Mobility-based caching strategy

Mobile users could move from one location to another or one wireless network to another network, which makes the users move from one edge server to another and one cache server to another. When users have not finished downloading the content from one cache server before moving to another location, the downloaded content may not be useful unless there is a mechanism to hand-off properly from one cache server to another [27,28]. Thus, based on the users' mobility trajectory, content should be cached to provide the best possible service with the least delay.

3.2.4. Hierarchical cooperative caching

Data can be cashed using a hierarchical framework to use the storage and caching effectively [29]. The work in [30] presents an interesting idea of cashing: use neighbors' storage space to cache the content, and use your own storage space, and/or strangers' data storage space for

caching.

3.2.5. Interest-based cooperative caching

Caching of the content can be done based on the interest of the content from users [31]. For instance, during night time residential, MEC could cache movies of certain types based on the population, such as kids movie if the given location has more kids and action movie if the given location has more adults who watch action movies very often.

3.3. Caching eviction/replacement strategy

We cannot cache everything all the time. We need to replace the old contents with new ones in the caches. The following sections provide some approaches for cache replacement strategies.

3.3.1. First In First Out (FIFO) replacement

FIFO is one of the easiest, fairest and popular strategies for content replacement in cache systems [32]. The content-first cached will be out first, and vice versa.

3.3.2. Least Recently Used (LRU) replacement

LRU approach replaces the content which is not used lately or which is not popular content in recent usage [33]. This method helps use the storage space effectively while meeting the demand of the end users.

3.3.3. Least Frequently Used (LFU) replacement

The LFU approach replaces the content, which is not popular or which is not used often by the new caches [32]. This approach does not remove any content if the incoming content is less popular than the content in the cache.

3.3.4. Time-Aware Least Recent Used (TLRU) replacement

TLRU is the advanced form of the LRU approach for replacing the cached contents [34] where Time-To-Use (TTU) is used to time stamp the content to see how often the content is used. TTU provides more options to decide which cache should be kept and which cache should be replaced in a timely manner.

3.3.5. Frequency-Based-FIFO (FB-FIFO) replacement

In FB-FIFO, variable-size protected segments are created and cached in the server [35]. Then based on the usage pattern, contents are replaced using FIFO manner. This approach is more effective than FIFO among others [35].

3.3.6. Aging popularity-based caching replacement

Based on the age and the popularity of the cached content, the old aged or least popular content is replaced by new content [36]. In this approach, the aging key value must be updated periodically to track the change of content popularity effectively.

3.3.7. Adaptive Replacement Cache (ARC)

ARC tracks both frequently used and recently used contents, as well as the removed history of both to replace the caching content [37]. ARC is considered to be outperforming the LRU among others [37].

4. Caching system behavior/performance and network optimizations

4.1. Caching system behavior/performance

Caching system behavior and performance are dependent on caching policies and caching replacement approaches, as discussed in previous sections. There is no single standard approach that fits all applications and needs of different users. While determining which caching approach is a better fit for a given scenario or application, we need to consider all features and characteristics so that we can support the application, then choose the best approach for content caching and cache replacement strategies. There are several models in the state of the artwork that analyze caching behavior and performance (e.g., [18,38–40]). For instance, the work in [39] used the Markov chain model to study the behavior of the caching system. The work in [40] studied the caching system using a discrete-time Markov chain. The work in [22] presented the policy structure of the Markov chain as a replacement of the LRU approach. The work in [41] used the stationary Markov model while studying the user mobility for caching using real trace information of mobile users.

4.2. Caching networks optimization models

There have been several studies for caching network optimization. In [42], the software-defined networking-enabled caching was studied for wireless networks. The work in [43] studied the energy-conscious caching in a wireless ad hoc network used to get the desired trade-off performance between access latency and energy utilization. Cooperative caching was studied in [44] to minimize the expected delay and maximize the overall system performance. A clustering-based caching plan for wireless divergent network systems has been studied in [45] to enhance the performance.

5. Comparison among edge, fog and cloud computing

Edge computing is a type of cloud computing where the computation or processing power is pushed out to be handled by the edge devices.

The Internet of Things (IoT) is expected to use edge computing to reduce latency and then use the power of computing through offloading. IoT, which collects huge data but has the limited capacity with individual devices, uses cloud computing in general to analyze the data. However, cloud computing introduces longer delays than edge computing. In order to minimize the bandwidth consumption and data transfer delay, edge computing is a better option than cloud computing.

These three layers, as shown in Fig. 2, can be interconnected by using gateways. We compare the latency of these three layers using numerical results obtained from simulations. The simulation configuration was similar to the ones given in [39]. We assumed that the number of edge nodes was 100, the number of fog servers was 10, and the number of cloud servers was 5 with computing capability of 10 GHz.

We considered both computation delay, as shown in Fig. 3 and communication delay, as shown in Fig. 4. When cloud computing is used, there will be most delay compared with fog and edge computing, as shown in Fig. 3. When edge computing is used, there will be least delay compared with fog and cloud computing, as shown in Fig. 3.

Similar observations were noted for communication delays (edge offers the least delay and the cloud offers the most delay), as shown in Fig. 4.

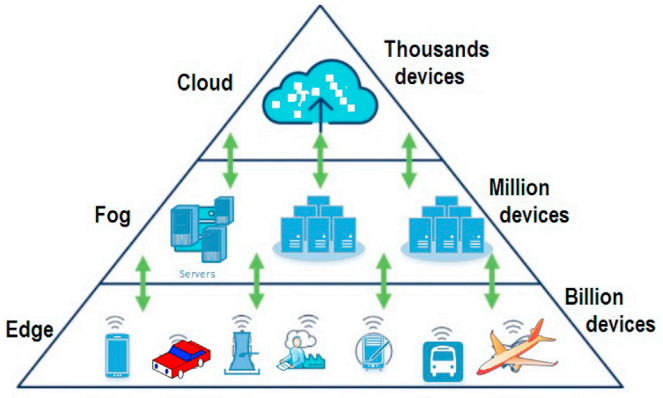

Fig. 2. Layered block diagram for comparing edge, fog and cloud computing in terms of number of devices and locations in the hierarchy.

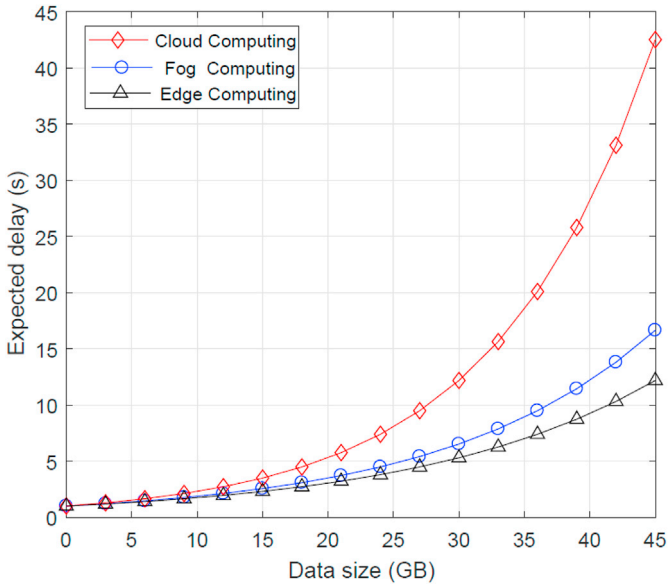

Fig. 3. Comparison of expected computation delays when data is offloaded to edge, fog and cloud for processing and receiving the response back to end device.

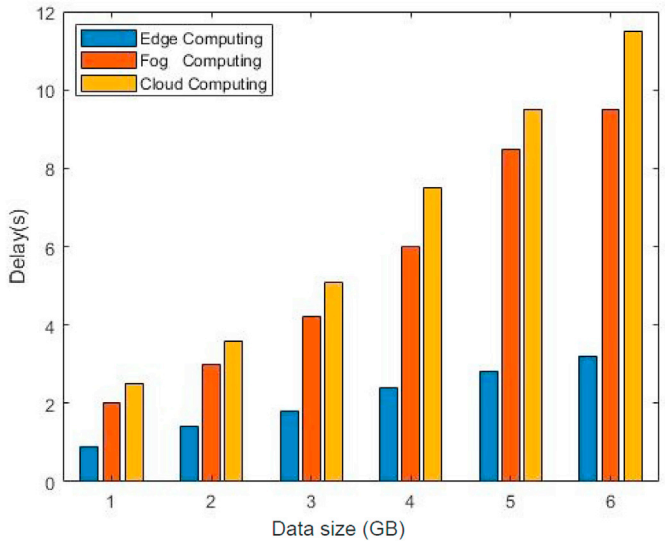

Fig. 4. Comparison in terms of communication delay when end users communicate with the edge, fog and cloud computing units.

6. Open research challenges

This section presents some of the research challenges and directions in mobile edge computing with content caching.

6.1. Heterogeneity

Wireless mobile networks are highly heterogeneous in terms of their users, devices used to access wireless services, wireless networks interfaces, and so on. End devices use different interfaces to access 3G, 4G, 5G, and WiFi. In such diverse wireless networks and heterogeneous bands, it is challenging to design/find a generic approach for hopping from one band to another for end user devices in MEC-enabled wireless systems.

6.2. User mobility

The mobility of the end users causes frequent disconnections with edge networks in MEC systems. When the devices are moving around, the overall performance significantly degrades. One of the challenging problems is to find an optimal solution to handle mobility in MEC-enabled wireless networks.

6.3. Pricing policy

User mobility is common in MEC networks. Due to the heterogeneity of networks, it is challenging to have generic pricing for usage charges. So, developing a dynamic pricing policy is one of the challenges in MEC services.

6.4. Scalability

The scalability property of caching and edge computing offers a high availability of the network services to any number of devices. However, it is challenging to meet the demand of the exponentially growing IoT devices. In order to reduce network bottleneck problems and service interruptions, the issue of scalability should be addressed in MEC networks.

6.5. Security

MEC servers could offer better security and privacy compared with cloud servers since they are closer to the end users. Whereas in cloud computing, the users might not have any idea where the data is stored. Still, securing the edge server is challenging.

6.6. Standard protocol

The standardization of edge computing could be a method to create an open environment for all, including research fellows and industries as well. As a new approach, MEC has not been standardized and appropriately implemented, and this creates lots of problems. The research and development for standardization could help expedite the widespread deployment of MEC systems for edge computing and content sharing.

6.7. Simulation platform

The standard simulation platform for evaluating the MEC platforms could help to model a real-world system. It could help design and evaluate the entire edge computing system and its feasibility without implementing the model in real time [45] by investing actual funds for edge infrastructures. The design of a universal simulation platform is also one of the open challenges.

7. Summary

In this survey paper, we have presented a survey on recent advances in mobile edge computing and content caching in edge servers. We have summarized not only several approaches of edge computing and content caching but also different issues of edge computing and caching and cache replacement strategies that aim to improve the end-user's quality of experience in terms of reduced latency and high throughput. We have also presented some open challenges and future research directions on the topic.

Acknowledgment

This work is partly supported by the US NSF under grants CNS 1650831, and HRD 1828811, and by the U.S. Department of Homeland Security under grant DHS 2017-ST-062-000003. However, any opinions, findings, and conclusions or recommendations expressed in this

document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the funding agencies.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.dcan.2019.08.004.

References

[1] Y. Mao, C. You, J. Zhang, K. Huang, K.B. Letaief, A survey on mobile edge computing: the communication perspective, IEEE Commun. Surv. Tutorials 19 (4) (2017) 2322–2358.
[2] M.T. Beck, M. Werner, S. Feld, S. Schimper, Mobile edge computing: a taxonomy, in: Proc. Of the Sixth International Conference on Advances in Future Internet, Citeseer, 2014, pp. 48–55.
[3] Y. Li, J. Liu, B. Cao, C. Wang, Joint optimization of radio and virtual machine resources with uncertain user demands in mobile cloud computing, IEEE Trans. Multimed. 20 (9) (2018) 2427–2438.
[4] R.P. Pradhan, G. Mallik, T.P. Bagchi, M. Sharma, Information communications technology penetration and stock markets–growth nexus: from cross country panel evidence, Int. J. Serv. Technol. Manag. 24 (4) (2018) 307–337.
[5] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, W. Wang, A survey on mobile edge networks: convergence of computing, caching and communications, IEEE Access 5 (2017) 6757–6779.
[6] M. Gregori, J. Gómez-Vilardebó, J. Matamoros, D. Gündüz, Wireless content caching for small cell and d2d networks, IEEE J. Sel. Area. Commun. 34 (5) (2016) 1222–1234.
[7] M. Agiwal, A. Roy, N. Saxena, Next generation 5g wireless networks: a comprehensive survey, IEEE Commun. Surv. Tutorials 18 (3) (2016) 1617–1655.
[8] A. Osseiran, F. Boccardi, V. Braun, K. Kusume, P. Marsch, M. Maternia, O. Queseth, M. Schellmann, H. Schotten, H. Taoka, et al., Scenarios for 5g mobile and wireless communications: the vision of the metis project, IEEE Commun. Mag. 52 (5) (2014) 26–35.
[9] P.K. Agyapong, M. Iwamura, D. Staehle, W. Kiess, A. Benjebbour, Design considerations for a 5g network architecture, IEEE Commun. Mag. 52 (11) (2014) 65–75.
[10] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan, Y. Zhang, Energy-efficient offloading for mobile edge computing in 5g heterogeneous networks, IEEE Access 4 (2016) 5896–5907.
[11] Y. Mao, J. Zhang, K.B. Letaief, Dynamic computation offloading for mobile-edge computing with energy harvesting devices, IEEE J. Sel. Area. Commun. 34 (12) (2016) 3590–3605.
[12] K. Zhang, Y. Mao, S. Leng, A. Vinel, Y. Zhang, Delay constrained offloading for mobile edge computing in cloud-enabled vehicular networks, in: 2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM), IEEE, 2016, pp. 288–294.
[13] X. Chen, L. Jiao, W. Li, X. Fu, Efficient multi-user computation offloading for mobile-edge cloud computing, IEEE/ACM Trans. Netw. 24 (5) (2016) 2795–2808.
[14] K. Habak, M. Ammar, K.A. Harras, E. Zegura, Femto clouds: leveraging mobile devices to provide cloud service at the edge, in: 2015 IEEE 8th International Conference on Cloud Computing, IEEE, 2015, pp. 9–16.
[15] L. Tianze, W. Muqing, Z. Min, Consumption considered optimal scheme for task offloading in mobile edge computing, in: 2016 23rd International Conference on Telecommunications (ICT), IEEE, 2016, pp. 1–6.
[16] M. Chen, Y. Hao, M. Qiu, J. Song, D. Wu, I. Humar, Mobility-aware caching and computation offloading in 5g ultra-dense cellular networks, Sensors 16 (7) (2016) 974.
[17] H. Ahlehagh, S. Dey, Video-aware scheduling and caching in the radio access network, IEEE/ACM Trans. Netw. 22 (5) (2014) 1444–1462.
[18] L. Breslau, P. Cao, L. Fan, G. Phillips, S. Shenker, et al., Web caching and zipf-like distributions: evidence and implications, in: Ieee Infocom, vol. 1, INSTITUTE OF ELECTRICAL ENGINEERS INC (IEEE), 1999, pp. 126–134.
[19] X. Wang, M. Chen, T. Taleb, A. Ksentini, V.C. Leung, Cache in the air: exploiting content caching and delivery techniques for 5g systems, IEEE Commun. Mag. 52 (2) (2014) 131–139.
[20] Y. Li, C. Liao, Y. Wang, C. Wang, Energy-efficient optimal relay selection in cooperative cellular networks based on double auction, IEEE Trans. Wirel. Commun. 14 (8) (2015) 4093–4104.
[21] J. Gu, W. Wang, A. Huang, H. Shan, Proactive storage at caching-enable base stations in cellular networks, in: 2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), IEEE, 2013, pp. 1543–1547.
[22] E. Baştuğ, M. Bennis, M. Kountouris, M. Debbah, Cache-enabled small cell networks: modeling and tradeoffs, EURASIP J. Wirel. Commun. Netw. 2015 (1) (2015) 41.
[23] B. Bai, L. Wang, Z. Han, W. Chen, T. Svensson, Caching based socially-aware d2d communications in wireless content delivery networks: a hypergraph framework, IEEE Wirel. Commun. 23 (4) (2016) 74–81.
[24] B. Chen, C. Yang, G. Wang, Cooperative device-to-device communications with caching, in: 2016 IEEE 83rd Vehicular Technology Conference (VTC Spring), IEEE, 2016, pp. 1–5.
[25] M. Zhang, H. Luo, H. Zhang, A survey of caching mechanisms in information-centric networking, IEEE Commun. Surv. Tutorials 17 (3) (2015) 1473–1499.
[26] R. J. Defouw, A. Sutton, R. W. Korngiebel, Caching method for selecting data blocks for removal from cache based on recall probability and size, uS Patent 6,742,084 (May 25 2004).
[27] T. Wei, L. Chang, B. Yu, J. Pan, Mpcs: a mobility/popularity-based caching strategy for information-centric networks, in: 2014 IEEE Global Communications Conference, IEEE, 2014, pp. 4629–4634.
[28] B. Cao, L. Zhang, Y. Li, D. Feng, W. Cao, Intelligent offloading in multi-access edge computing: a state-of-the-art review and framework, IEEE Commun. Mag. 57 (3) (2019) 56–62.
[29] M.R. Korupolu, C.G. Plaxton, R. Rajaraman, Placement algorithms for hierarchical cooperative caching, J. Algorithms 38 (1) (2001) 260–302.
[30] Y. Wang, J. Wu, M. Xiao, Hierarchical cooperative caching in mobile opportunistic social networks, in: 2014 IEEE Global Communications Conference, IEEE, 2014, pp. 411–416.
[31] J. Iqbal, P. Giaccone, Interest-based cooperative caching in multi-hop wireless networks, in: 2013 IEEE Globecom Workshops (GC Wkshps), IEEE, 2013, pp. 617–622.
[32] V. Martina, M. Garetto, E. Leonardi, A unified approach to the performance analysis of caching systems, in: IEEE INFOCOM 2014-IEEE Conference on Computer Communications, IEEE, 2014, pp. 2040–2048.
[33] J. Liu, G. Wang, T. Huang, J. Chen, Y. Liu, Modeling the sojourn time of items for in-network cache based on lru policy, China Commun. 11 (10) (2014) 88–95.
[34] M. Bilal, S.-G. Kang, Time aware least recent used (tlru) cache management policy in icn, in: 16th International Conference on Advanced Communication Technology, IEEE, 2014, pp. 528–532.
[35] H. Gomaa, G.G. Messier, C. Williamson, R. Davies, Estimating instantaneous cache hit ratio using Markov chain analysis, IEEE/ACM Trans. Netw. 21 (5) (2013) 1472–1483.
[36] Z. Ming, M. Xu, D. Wang, Age-based cooperative caching in information-centric networking, in: 2014 23rd International Conference on Computer Communication and Networks (ICCCN), IEEE, 2014, pp. 1–8.
[37] G. Jia, G. Han, J. Jiang, L. Liu, Dynamic adaptive replacement policy in shared last-level cache of dram/pcm hybrid memory for big data storage, IEEE Trans. Ind. Inf. 13 (4) (2017) 1951–1960.
[38] A.V. Aho, P.J. Denning, J.D. Ullman, Principles of optimal page replacement, J. Assoc. Comput. Mach. 18 (1) (1971) 80–93.
[39] E.J. Rosensweig, J. Kurose, D. Towsley, Approximate models for general cache networks, in: 2010 Proceedings IEEE INFOCOM, IEEE, 2010, pp. 1–9.
[40] E.J. Rosensweig, D.S. Menasche, J. Kurose, On the steady-state of cache networks, in: 2013 Proceedings IEEE INFOCOM, IEEE, 2013, pp. 863–871.
[41] J. Zhang, X. Zhang, W. Wang, Cache-enabled software defined heterogeneous networks for green and flexible 5g networks, IEEE Access 4 (2016) 3591–3604.
[42] W. Li, E. Chan, D. Chen, Energy-efficient cache replacement policies for cooperative caching in mobile ad hoc network, in: 2007 IEEE Wireless Communications and Networking Conference, IEEE, 2007, pp. 3347–3352.
[43] X. Li, X. Wang, S. Xiao, V.C. Leung, Delay performance analysis of cooperative cell caching in future mobile networks, in: 2015 IEEE International Conference on Communications (ICC), IEEE, 2015, pp. 5652–5657.
[44] M.S. ElBamby, M. Bennis, W. Saad, M. Latva-Aho, Content-aware user clustering and caching in wireless small cell networks, in: 2014 11th International Symposium on Wireless Communications Systems (ISWCS), IEEE, 2014, pp. 945–949.
[45] B. Sampathkumar, G. Yongkun, Cloud computing simulator, uS Patent App. 14/983,765 (Jul. 6 2017).