Journal of Multimedia and Information System Vol. 2, No 1, March 2015(pp. 143-152): ISSN 2383-7632(Online)
http://dx.doi.org/10.9717/jmis.2015.2.1.143

Distributed Multimedia Scheduling in the Cloud

Mengting Zheng[1,2,*], Wei Wang[2]

Abstract

Multimedia services in the cloud have become a popular trend in the big data environment. However, how to efficiently schedule a large number of multimedia services in the cloud is still an open and challengeable problem. Current cloud-based scheduling algorithms exist the following problems: 1) the content of the multimedia is ignored, and 2) the cloud platform is a known parameter, which makes current solutions are difficult to utilize practically. To resolve the above issues completely, in this work, we propose a novel distributed multimedia scheduling to satisfy the objectives: 1) Develop a general cloud-based multimedia scheduling model which is able to apply to different multimedia applications and service platforms; 2) Design a distributed scheduling algorithm in which each user makes a decision based on its local information without knowing the others' information; 3) The computational complexity of the proposed scheduling algorithm is low and it is asymptotically optimal in any case. Numerous simulations have demonstrated that the proposed scheduling can work well in all the cloud service environments.

Key Words: multimedia services, cloud computing, distributed scheduling, performance analysis.

I. INTRODUCTION

Nowadays, cloud computing has become the most popular computing method in the big data environment. Meanwhile, multimedia applications, e.g., video communications, online game, *etc.*, have dominated the service content of the cloud [1-2]. Usually, multimedia applications cost substantial computational resource, while the computational resource of the cloud is limited as well. Thus, a natural question arises: how efficiently schedule these multimedia applications in the cloud to maximize the resource utilization? That is, how to assign appropriate resource to the appropriate multimedia applications? This is a very important issue for the applications of media cloud.

Recently, a number of scheduling algorithms have been proposed in the framework of media cloud [1-6]. However, none of them has successfully addressed the above question (the detailed introduction please see Section II). Existing methods usually assume that the parameters of the cloud are known to the scheduler or the contents of the multimedia applications which do not impact the scheduling performance [7]. Although the above assumptions can simplify problematic formulation and the corresponding solution, it is impractical due to the following reasons: 1) the cloud parameters are usually blind to the users. Even if the user can obtain the parameters in some way, the parameters are usually dynamic. That is, they will change over the time, which completes the parameter obtaining procedure. 2) As studied in the traditional wireless multimedia communications, multimedia contents have played a very important role, and they significantly impact on the scheduling performances. Unfortunately, current studies ignore the contents of the multimedia, therefore, the scheduling is not optimal in general [8-9].

Essentially, the above two issues raise several fundamental problems: 1) how to realize the multimedia scheduling without the knowledge of the cloud parameters and the other users' information? 2) How to provide a general model to include the impact of the multimedia contents. 3) How to guarantee the scheduling scheme can be applied in different cases (heavy traffic and light traffic) with an online implementation. In fact, the above three points establish the main technical challenge for the multimedia scheduling in the cloud, and existing works cannot resolve these questions well [9].

In this work, we propose a novel distributed cloud-based multimedia scheduling which jointly resolves the above challenges. As far as I know, this work is the first one to

Manuscript received February 17, 2015; Revised March 1, 2015; Accepted April 18, 2015. (ID No. JMIS-2015-0006)
Corresponding Author(*): Mengting Zheng, E-mail: zhengmengting@gmail.com
[1]Department of EECS, UCLA, CA, USA, E-mail: zhengmengting@gmail.com
[2]Department of CS, Anhui University of Technology, Anhui, China, E-mail: huhui19650501@163.com

provide a systematic solution across different platforms. In summary, the contributions of this paper are as follows.

- Develop a general cloud-based multimedia scheduling model which is able to apply to different multimedia applications and service platforms.
- Design a distributed scheduling algorithm in which each user makes a decision based on its local information without knowing the others' information;
- The computational complexity of the proposed scheduling algorithm is low and it is asymptotically optimal in any case.

The remainder of this paper is organized as follows. Section II overviews the related work on the cloud media. Then, the service model is described and the mathematical problems are formulated Section III. In Section IV, we design a distributed multimedia scheduling scheme and analyze its characteristics. Subsequently, substantial simulations are conducted in Section V to validate the proposed scheme. At last, Section VI concludes the paper with a summary.

II. RELATED WORKS

The multimedia scheduling in the cloud is a very popular topic in recent years [1-6], and existing methods can be roughly divided into two categories. The first one is the scheduling scheme with known cloud parameters [7], for example, average service time, waiting time, user QoE model, etc. Typically, these methods are based on approximations of user demand, service time, and waiting time. In addition, there are a lot of works [6], [7] using the approximation of multimedia servers to achieve the near-optimal solution. In particular, [8], [9] make advantage of approximating the user service demand to investigate a few multimedia applications scheduling under strict service time constraint; [1], [3], [10] deal with several multimedia applications with different QoS requirements by jointly averaging the user demand and service time. In addition, a number of approximation methods also consider the scheduling problem with multiple performance metrics (e.g., [11] and the references therein). These works are important to understand the role of the system parameters, and they provide a series of solutions to simplify the multimedia scheduling problem in the media cloud.

The second category mainly deals with the case of the heavy traffic, since it provides a simple problem formulation environment to describe [12]. However, in reality, there are other cases, such as light traffic. For a practical cloud, in the rush time the traffic is usually heavy traffic, while in the night, the traffic is always light. Since the multimedia servers are operated all the time, it is necessary to study both the heavy and light traffics. In terms of the light traffic, the impact of user request uncertainty is first investigated by [13], which provides the relationship between the waiting time and user demand. However, it does not provide a systematic scheduling algorithm to optimize the performance. Subsequently, a number of literature concentrate on the performance analysis in the framework of random demand rates and service times, does not consider how to achieve the optimal solution [6]. Most recently, [4] designs an efficient scheduling approach with unknown service rate of the server. However, this work assumes that the user demand satisfies some specific model which is a known parameter for the scheduler. Lai [3] proposes a general estimation model to estimate the user demand and service time based on the large-scale historic data. However, sometimes historic data is very difficult to obtain, and it usually costs a lot of learning and processing time, which is very hard to apply to real-time scenario. The above all studies deal with the heavy traffic case only, and ignore the case of light traffic. In fact, the multimedia scheduling in the terms of light traffic is also complex in reality [13].

In contrast, our work has several different aspects. 1) We propose a totally blind distributed multimedia scheduling schemes in the cloud, in which the corresponding parameters are unknown to the scheduler. In particular, we use a new formulation to describe the distributed scheduling scenario; 2) The proposed scheduling scheme is able to apply to both heavy traffic and light traffic cases, and work well in both cases; 3) The proposed solution possesses the property of asymptotic optimality for various unknown system parameters; 4) The computational complexity of the proposed scheme is so simple that it can be implemented in an online manner.

III. SYSTEM MODEL

In this work, we consider a classical media cloud, in which there are U user classes and S server classes. The user and server in the same class has the same parameters. The media demand rate for user u $(u \leq U)$ is denoted by P_u, and the total demand rate is $\sum_{u=1}^{U} P_u$. In terms of the service time, it is usually independent with each class of the server s $(s \leq S)$. The service process can be viewed as a traditional queue procedure with Poisson rate, so the service time is exponential, and the average service time of a user served by a class s is $1/\mu_s$. Assume that there are N_s servers in server class s, and the total number of the server is $\sum_{s=1}^{S} N_s$. Moreover, let ξ be a specific scheduling, and $W_u(\xi)$ be the waiting time of the user class u.

In this work, we jointly consider both waiting time and service time to maximize the cloud performance. To this end, the designed scheduling scheme should be very delay-sensitive, and robust enough to deal with different traffic cases. Motivated by [4], we introduce Y_s to represent the minimum free time of server class s, and $f_s = Y_s/\sum_{s=1}^{S} Y_s$ to show the average free time factor. In this work, we employ a new average free time factor compared to [5], that because of the following reasons. 1) Y_s is always difficult to obtain in a practical media cloud, in particular in the dynamic environment. 2) The average value of Y_s for all the users cannot truly describe the state of the server. To resolve this problem, we introduce the service window T to denote the service period of the server on average. In fact, the value of T plays the important role in determining the optimal scheduling, since it depends on the service fashion. Previous works always assume that the value of T is a known parameter, however, in this work, we treat it is unknown to the scheduler which is much more practical.

The unknown parameter T introduces some difficult problem, in particular, the traditional convex optimization formulation is not available. To resolve this difficulty, motivated by [4], we use the probability formulation in which all the variables and constraints are formulated by the probability. Actually, this expression is practical in the media cloud, since all the service quality cannot be provided by an accurate expression (e.g., dynamic system, etc.), but can be formulated with the probability expression. In this case, the core problem is that how to include all the cases in the probability.

To jointly consider the heavy and light traffics, we provide a probability-based optimization problem. The scheduler first subjectively selects the service window T, then choose the most suitable user to the server satisfying

$$\begin{gathered} min\ \sum_{u=1}^{U}\sum_{s=1}^{S} P_u W_u(\xi;\mathrm{T})/\mu_s \qquad (1) \\ \text{s.t. } N_s/\sum_{s=1}^{S} N_s \le f_s, \end{gathered}$$

where $P_u W_u(\xi;\mathrm{T})$ denotes the total waiting time in the service period T. Unfortunately, the goal function (1) is not a classical convex or concave function, we should propose an approximation method to establish a new function. It is well known that the probability-based function can flexibly adapt to the estimation error that the scheduler can tolerate.

Essentially, the core part of resolving the problem (1) is to control the estimation error of the user demand and service time given a service time in the tolerable extent. If so, we can obtain the desired asymptotical optimality at all the cases. As discussed previously, the above goal can be satisfied when the traffic is large (heavy traffic). However, how to deal with the light traffic, that is, the arrival rate of the multimedia demand is not enough or the server has some idle time to deal with the multimedia demand, is still an open problem. By the definition, the solution of the heavy traffic is totally different from the light traffic due to the following reasons. 1) In the heavy traffic, there is a strong relationship between the service time and waiting time, while this relationship is not clear at the light traffic; 2) the estimation error of the system parameter is much easier in the case of the heavy traffic, that is because server idle time is ignorable in this case; 3) User demand in the heavy traffic can be estimated in a relatively simple method. That is because the total amount of the user demand in this case is much larger than the light traffic case.

Additionally, we also seek the asymptotical optimality solution for all the cases. If the service time is set appropriately, or the estimation error falls within some extent, we can achieve the asymptotical optimality in theory. So, the core problem is how to set the error extend, and how to guarantee this error. Here we propose a new strategy for dealing with the error problem. Specifically, if one server is idle, it only deals with the demand from his neighbor, and follows the first-in-first-out rule. In this case, the most advantage of this strategy is to avoid the error propagation event which dominates the main error estimation. It should be noted that in a short time period, the asymptotically optimal allocation cannot be provided since the service window is too short. Here we formally define that the asymptotical optimization means the gap between the approximation and original problem tends to zero when the service time goes to infinity. Basically, this idea comes from the following two observations: 1) the estimation error can be reduced once a simple function with a few variables is employed, and 2) the proposed approximated function can well fit the user demand rate and service time in both heavy and light traffics.

IV. DISTRIBUTED SCHEDULING

1. Multimedia Scheduling Framework

Similar to the traditional cloud computing architecture, the multimedia scheduling framework can be viewed a distributed queue processing system. We can divide this framework into four layers based on the functions:

- *Information Collection (IC) Layer*. This layer provides the information collection software, in which all the information of the served multimedia applications and users are collected for future processing. In fact, this layer acts as the basis for the future estimation for the system parameter, since all profiles of the user and multimedia applications are crucial for system parameters. Basically, this information collected by the local information is handled by the user itself, so it is possible for distributed implementation in the future.
- *Information Relationship (IR) Layer*. IR locates at the top of the IC layer, and its main function is to map the collected information to the system parameters. From the perspective of the function, IR acts as the bridge between the physical layer and the application layer. Specifically, IR includes the information processing, information retrieve, and information mapping. Although this layer can be omitted if the cloud in the light traffic, however, it is very useful at the heavy traffic since it can reduce the computational complexity significantly [14].
- Virtual Computing (VC) Layer. VC works based on IC and IR layers. VC provides a unified computing platform for all the scheduling, including estimating the system parameters, and revising the estimation method accordingly. In fact, the computing methods are quite common in some specific platforms, however, we provide a general one which is suitable for both video, audio, and data services.
- *Data Control (DC) Layer*. DC plays the core role in the distributed scheduling, since it deals with the data from the service request, service surveillance, and service process. Usually, DC also deals with the charge problem, and excludes the users with bad reputation and experience. Importantly, the scheduling algorithm should be embedded in this layer with the information of other layers.

However, the above four layers just serve as the basic framework for the multimedia scheduling framework. To make it possible, it is necessary to include more functions. Compared to traditional cloud platform, multimedia cloud has the follow specific features for the different functions:

- *Dynamic adaptation*. Usually, a large number of multimedia services are available in the cloud. Dynamic adaptation is an efficient manner for multimedia service and different users to satisfy their service demand in a dynamic media cloud environment.
- *Flexible service*. Since multimedia services in the cloud are so huge that the servers are able to deal with different kinds of service. In other word, it is necessary to provide a flexible service to realize the dynamic adaptation. In addition, the different multimedia application should easily handle unexpected service request without the change of the system framework.

Overall, the multimedia scheduling framework is shown in Figure 1.

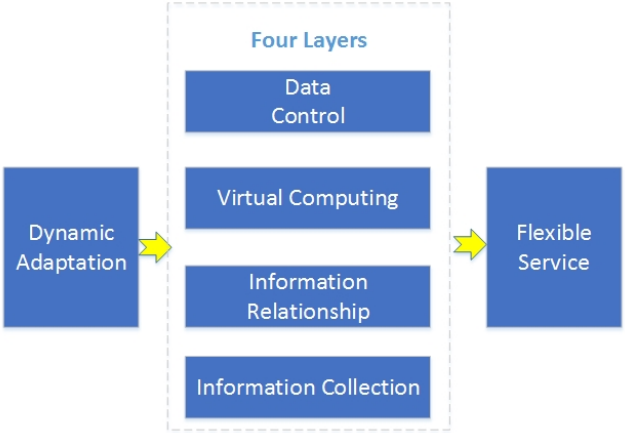

Fig. 1. Multimedia scheduling framework in the cloud.

2. Distributed Multimedia Scheme

Based on the designed multimedia scheduling framework, we can propose a distributed multimedia scheduling scheme across all the cloud computing platform. In this part, we mainly deal with four important issues:

1. How to accurately collect the information of mu-ltimedia applications and cloud?
2. How to assign the appropriate resource to the multiple multimedia applications?
3. How to ensure that the proposed scheme achieves the asymptotical optimality?
4. How to guarantee the proposed scheduling scheme is suitable for online implementation?

To accomplish these goals, we should allocate the available multimedia servers according to the average users' requests, and map the heterogeneous multimedia traffics based on some specific rule. We propose a *distributed multimedia scheduling algorithm* (DMSA) outlined in Fig. 2. Intuitively, the proposed DMSA can adapt to the different multimedia traffics. In particular, DMSA does not need the knowledge of the system parameters and the information of the other users. It only needs to calculate the optimal server and mapping rule at each time slot, as a result, the computational complexity is low. Thus, it is possible for online implementation.

In fact, the core part lies in how to approximate the service time and the waiting time, which significantly impact the system performance. In DMSA, we employ a linear dynamic method which is firstly designed by [3]. However, this method cannot be applied directly due to the following reasons:

1. The whole approximation process is based on the exact system parameter, and cannot deal with the approximation error.
2. The computational complexity is so huge that it cannot be implemented in an online manner.
3. The approximation error is not controlled, and the error propagation can be occurred when the traffic is not large (light traffic).

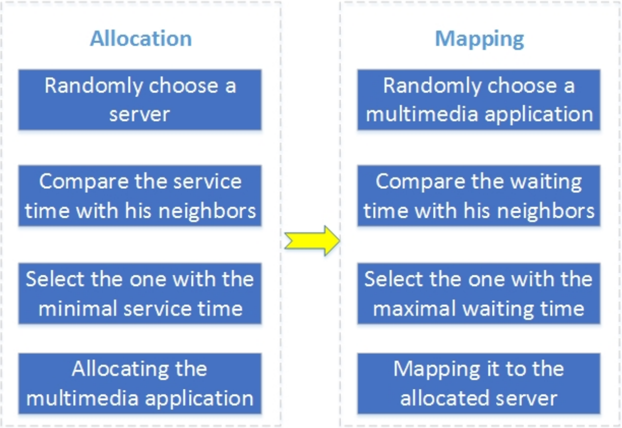
Allocation
Randomly choose a server
Compare the service time with his neighbors
Select the one with the minimal service time
Allocating the multimedia application
Mapping
Randomly choose a multimedia application
Compare the waiting time with his neighbors
Select the one with the maximal waiting time
Mapping it to the allocated server

Fig. 2. The architecture of DMSA.

In fact, the basic idea of the DMSA is similar to the traditional content of recommendation problem. Li and etc. [14] designs a content recommendation system for media cloud, which contains two fundamental components: 1) adaptive multimedia receiver, which is operated by the end user to receive the specific multimedia service. 2) Multimedia sharing scheme works if the service resources do not exist in the cloud. It should be noted that the connection fashion of the multimedia servers can be viewed as an independent service manner [15].

In our proposed DMSA, the core approximation algorithm also consists of several steps: starting, transmitting, ordering, and recommendation. In particular, at the beginning, one of the multimedia server is randomly selected as the initial value which is processed in the member of the server. Then, we need to record the multimedia server, which also consists of two parts: analysis and store. In terms of analysis, relationship analysis for each multimedia server is implemented. Specifically, the relationship degree between the every server is calculated. The service with the highest relationship gets the highest score, and the service with the lowest relationship gets the lowest score accordingly. After that, the scores are reported to the neighbor servers for store. Typically, the scores are ordered in each server, which is implemented by the server itself and does not impact the others. This implementation also contains two parts: the first one is to study the highest order in existing multimedia service and the second one is to map server with the highest score to the most appropriate multimedia service. At last, a recommendation of the multimedia service and the server is reported to the scheduler, which is implemented in each server. In this sense, the proposed DMSA is a totally distributed scheme.

The applications of the DMSA follow the basic first-come-first-service rule. In most cases, it is necessary to include different services into the scheduling scheme. The whole scheduling application is shown in Figure 3.

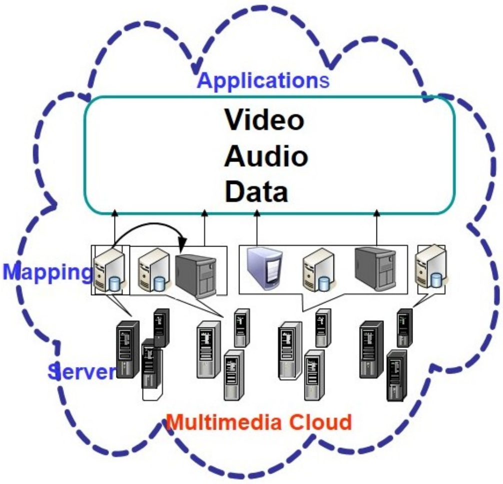

Fig. 3. Scheduling application in DMSA.

3. Performance Analysis

Usually, the end users pay more attention to the user experience which provides a more subjective evaluation in terms of satisfaction, reliability, delay, and jitter, *etc*. The multimedia applications in the proposed cloud-based multimedia scheduling require a certain user experience model to provide high flexibility and reliability, especially for some real-time multimedia applications [16].Specifically, 1) the scheduling algorithm ensures that the multimedia transmission requires a higher stability; 2) The allocated resource for each application can be utilized with high-probability reliability; 3) Some specific applications, e.g., file downloading, require low frequency data reporting and a relatively stable resource allocation while the user demand may need a much higher frequency of information mapping with more accurate information. In addition, in terms of some delay-sensitive applications, there are some variables to deal with the underlying protocols for error propagation. Due to the delay-sensitive feature of the multimedia applications, different cloud computing technologies usually provide diverse implementation methods which greatly impact the scheduling performance [17].

The core part of the proposed DMSA is the advanced information mapping technology which is designed to measure and analyze multimedia service and computing to pave the way for adaptive multimedia scheduling scheme. Given that multimedia traffic is a classical dynamic data, transmitting from the end user to the multimedia server, distributed scheduling is one of the main network congestion problems in which all the data may queue at the beginning of each multimedia encoding part.

For a specific multimedia application, demand estimation is a necessary step in response to the variations of heterogeneous multimedia traffics. The function of estimation error is to analyze the sources of the errors to reduce their impact corresponding to the system traffic. The scheduler can adjust the service time of the allocated server by reallocating the multimedia traffic. The multimedia applications are routed to the server with the larger handle capacity. In general, the error propagation can be removed if we introduce the revision step, in which all the estimations are recorded in the server, and the average one is used. It is important to note that the average value is calculated by the server when it receives the multimedia applications from the other servers. In practical, the error estimation algorithm cannot realize the above goal function at the few recorded traffic. In this case, we have two methods: 1) waiting more time to receive more multimedia traffic, or 2) using the historic data, which is also viewed as the recorded data.

To provide a general computing platform with intelligently scheduling function, we utilize a visual scheduling framework which divides the scheduling part into four components, *i.e.*, information report, server assignment, traffic management, and reliability guarantee. In addition, these four components can be jointly performed at the application layer.

V. SIMULATION RESULTS

In this section, we evaluate the performance of the distributed multimedia scheduling algorithm. In particular, all the simulations are based on real-world multimedia traces which contain three categories: video, audio, and data. We compare our DMSA with an ideal case, where the scheduler knows the user demand rate and server service time in advance, which can be viewed as the optimal solution. The simulation settings are: $N_s = 30$, $\mu_s = 1\text{ms}$, $U = 10$.

At first, we validate the asymptotic optimality of the proposed DMSA. Figure 4 and Figure 5 show the performance of our proposed DMSA versus that of the ideal case in the heavy traffic and light traffic, respectively. From the given results, we know that the performance gap between the two schemes is ignorable when the time goes largely. That is to say, DMSA is able to get the asymptotically optimal in both heavy and light traffics. Moreover, they also imply that when the processes of scheduling can be treated as two independent processes: allocation and mapping. This is very important for the media cloud, since this method can be extended to other platforms if the distributed scheduling is necessary. Moreover, we should note that the limit of the heavy and light traffics is very difficult to define in practical, and this work provides a unified expression, which is very valuable in general.

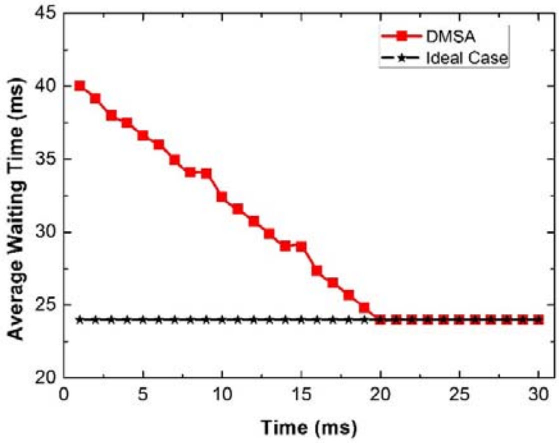

Fig. 4. Asymptotical optimality in heavy traffic.

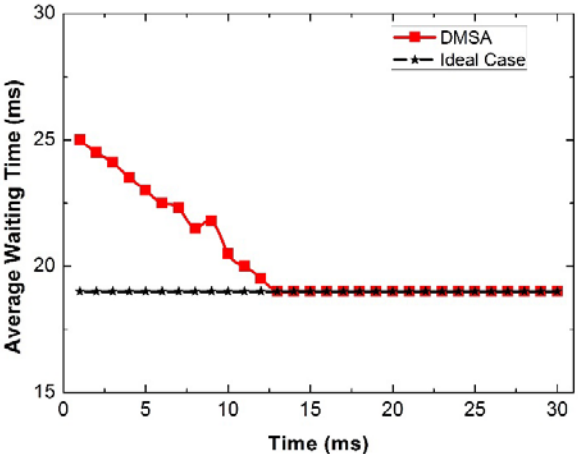

Fig. 5. Asymptotical optimality in light traffic.

Next, we validate the estimation of the waiting time and service time proposed in this work. To provide a comparable result, we introduce the conception of the performance gap, which denotes the loss due to the estimation. For simplicity, we normalized the performance matric. The results of the Figure 6-Figure 9 clearly demonstrate that the proposed approximation method is suitable for the cloud media. It should be noted that the heavy traffic and the light traffic is a relative conception. In our simulations, the traffic is determined by the arrival rate of the arriving multimedia applications. In particular, if the arrival rate is larger than or equal to the processing capacity of the server, we treat it as the heavy traffic, otherwise, it is a light traffic case. In fact there are other definitions on heavy and light traffics, and our proposed DMSA can be extended to other cases with a slight modification.

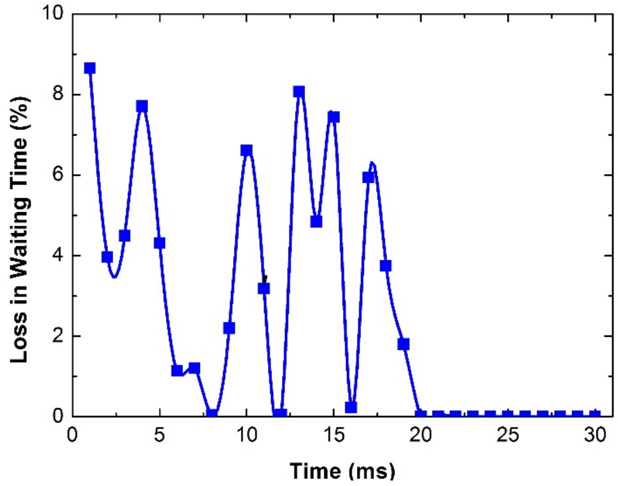

Fig. 6. The loss in waiting time at heavy traffic.

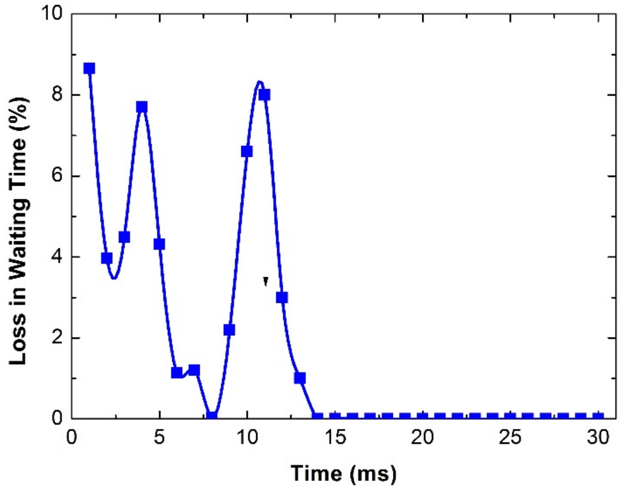

Fig. 7. The loss in waiting time at light traffic.

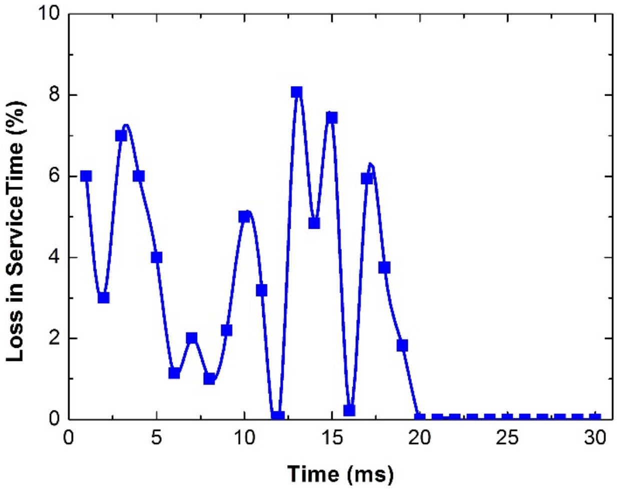

Fig. 8. The loss in service time at heavy traffic.

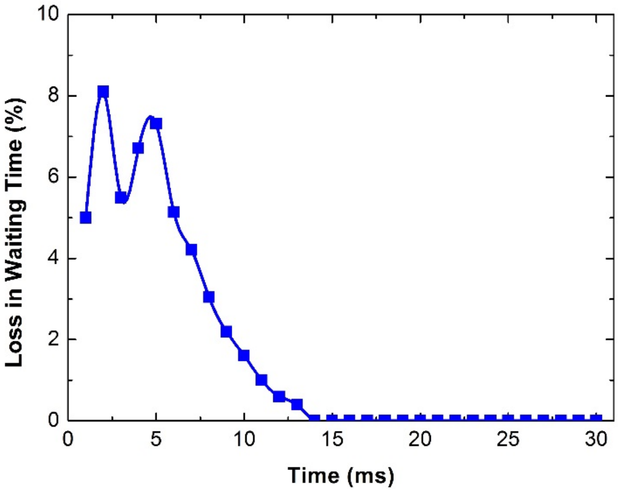

Fig. 9. The loss in service time at light traffic.

Subsequently, we examine the performance improvement of the proposed DMSA in other aspects. According to our theoretical analysis, our scheme can adaptively assign the multimedia applications in a dynamic environment. To test that, we compare our DMSA with existing blind scheduling scheme [5] in a dynamic environment, in which the servers can join and leave the data center randomly. Figure 10 and Figure 11 show the comparison results in terms of the waiting time. From the given results, we can find that the waiting time of the proposed DCMA has been reduced substantially. This point is consistent with our theoretical analysis. It should be noted that the calculation of the waiting time is the same for the two schemes, and the cloud platform is the same as well. To provide a practical dynamic environment, all the servers are placed in a distributed environment, and each server can join and leave randomly. Moreover, several servers can form a cluster, and join and leave the data center randomly.

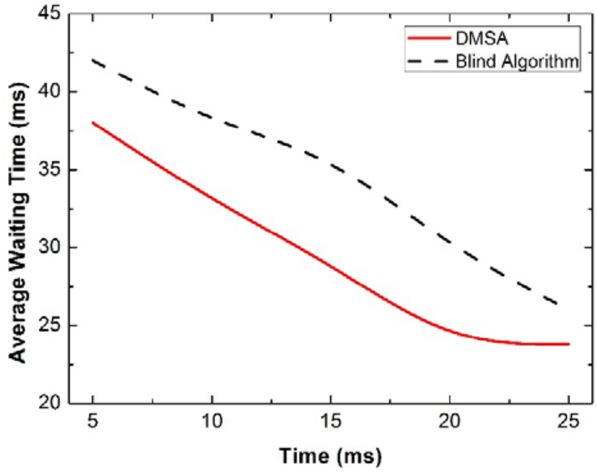

Fig. 10. Waiting time comparison at heavy traffic.

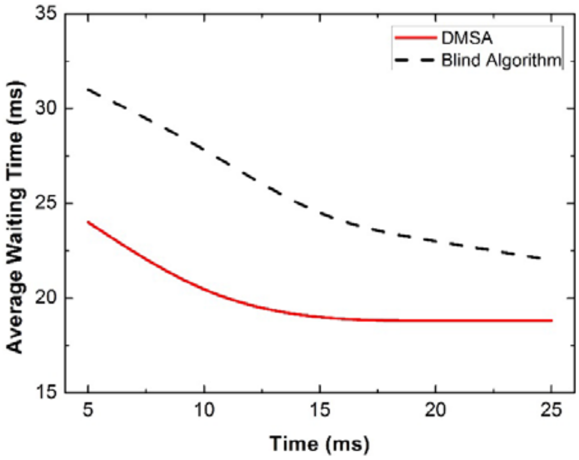

Fig. 11. Waiting time comparison at light traffic.

Finally, we compare our DMSA with the blind algorithm in terms of the average service time in the dynamic environment. Figure 12 and Figure 13 clearly show the advantages of our proposed scheme.

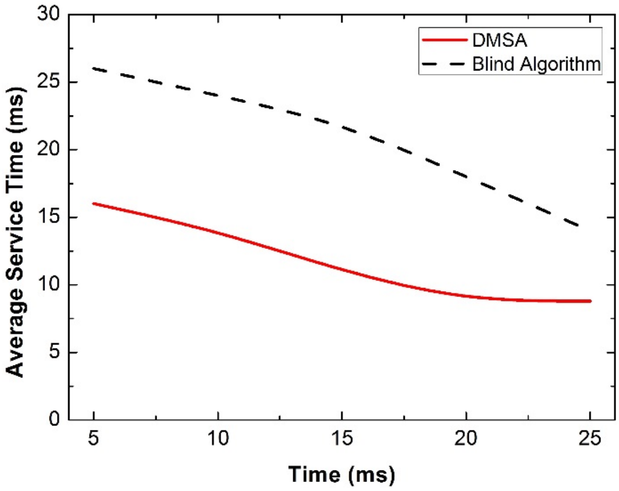

Fig. 12. Service time comparison at light traffic.

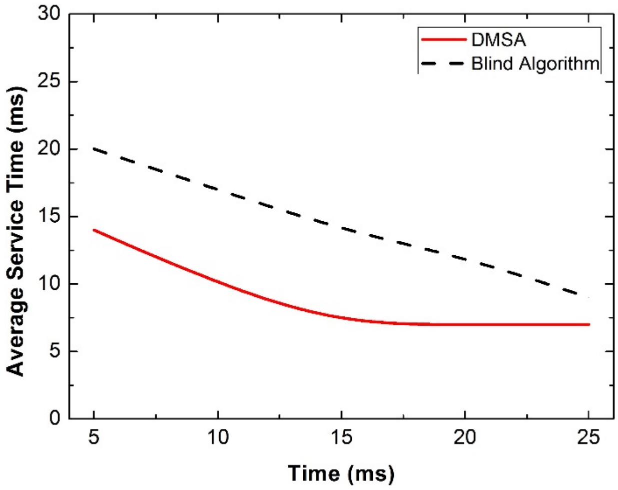

Fig. 13. Service time comparison at light traffic.

VI. CONCLUSIONS

In this work, we focus on the multimedia services in the cloud and deal with how to efficiently schedule a large number of multimedia services in the cloud. As pointed out in this work, current cloud-based scheduling algorithms exist two main problems: 1) the content of the multimedia content is ignored, and 2) the cloud platform is a known parameter, which make current solutions are difficult to utilize practically. To resolve these issues completely, in this work, we propose a novel distributed multimedia scheduling to satisfy the objectives. 1) Develop a general cloud-based multimedia scheduling model which is able to apply to different multimedia applications and service platforms; 2) Design a distributed scheduling algorithm in which each user makes a decision based on its local information without knowing the others' information; 3) The computational complexity of the proposed scheduling algorithm is low and it is asymptotically optimal in any case. Numerous simulation results have demonstrated that the proposed scheduling can work well in all the cloud service environments.

In our future work, we focus on extending DMSA to a larger scale of media cloud. As far as I know, although this work has dealt with the environment problem, there is still some open problems:

- How to reduce the operation complexity at the larger scale of the data center?
- How to design a general scheduling scheme embedded into current protocol without significant modification.

In our going work, we will study these questions carefully.

Acknowledgement

We thank Dr. Fen Tian, Dr. Rui Su, and Dr. Xi Tian for detailed and constructive discussion with us, and thank Dr. Feng Wu for providing us a lot of valuable comments and suggestions.

References

[1] W. Zhu, C. Luo, J. Wang, and S. Li, “Multimedia Cloud Computing,” *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 59-69, May 2011

[2] J. Rodrigues, L. Zhou, L. Mendes, K. Lin, and J. Lloret, “Distributed Media-Aware Flow Scheduling in Cloud Computing Environment”, *Computer Communications*, vol. 35, no. 1, pp. 1819-1827, September 2012.

[3] C. F. Lai, J. H. Chang, C. C. Hu, Y. M. Huang, H. C. Chao, “CPRS: A Cloud-Based Program Recommendation System for Digital TV Platforms,” *Future Generation Computer Systems*, vol. 27, no. 6, pp. 823-835, June 2011.

[4] L. Zhou, H. Wang, “Toward Blind Scheduling in Mobile Media Cloud: Fairness, Simplicity, and Asymptotic Optimality,” *IEEE Transactions on Multimedia*, vol. 15, no. 4, pp.735-746, June 2013.

[5] L. Zhou, Z. Yang, J. Rodrigues, and M. Guizani, “Exploring Blind Online Scheduling for Mobile Cloud Multimedia Services,” *IEEE Wireless Communications*, vol. 20, no. 3, pp. 54-61, June 2013.

[6] J. Rodrigues and P. Neves, “A Survey on IP-based Wireless Sensor Networks Solutions,” *International Journal of Communication Systems*, vol. 23, no. 8, pp. 963-981, 2010.

[7] W. W. Zhang, Y. G. Wen and D. P. Wu, “Collaborative Task Execution in Mobile Cloud Computing under Stochastic Wireless Channel,” *IEEE Transactions on Wireless Communications*, vol. 14, no.1, pp.81-93, January 2015.

[8] Y. C. Jin, Y. G. Wen, H. Hu and M. J. Montpetit, “Reducing Operational Costs in Cloud Social TV: an Opportunity for Cloud Cloning,” *IEEE Transactions on Multimedia*, vol. 16, no. 6, pp.1739-1751, October 2014.

[9] J. H. Tang, W. P. Tay and Y. G.Wen, “Dynamic Request Redirection and Elastic Service Scaling in Cloud-Centric Media Networks,” *IEEE Transactions on Multimedia*, vol. 16, no. 5, pp.1520-1445, August 2014.

[10] H. Hu, Y. G. Wen, H. B. Luan, T. S. Chua and X. L. Li, “Towards Multi-Screen Social TV with Geo-Aware Social Sense,” *IEEE Multimedia*, vol.21, no. 3, pp.10-19, July-September 2014.

[11] M. Chen, J. Wan, S. Gonzalez, X. Liao, V. Leung, “A Survey of Recent Developments in Home M2M Networks,” *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 98-114, Feb. 2014.

[12] M. Chen, Y. Zhang, Y. Li, S. Mao, V. Leung, “EMC: Emotion-aware Mobile Cloud Computing in 5G”, *IEEE Network*, vol. 29, no. 2, pp.32-38, March-April 2015.

[13] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, “A Survey on Smart Grid Communication Infrastructures: Motivations, Requirements and Challenges”, *IEEE Communications Surveys and Tutorials*, vol.15, no. 1, pp.5-20, Feb. 2013.

[14] Q. Li, Y. Xu, RQ Hu, and Y. Qian, “Optimal Fractional Frequency Reuse and Power control in the Heterogeneous Wireless Networks”, *IEEE Transactions on Wireless Communications*, vol.12, no.6, pp.2658- 2668, June 2013.

[15] J. Xiao, R. Q. Hu, Y. Qian, L. Gong, and B. Wang, “Expanding LTE Network Spectrum with Cognitive Radios: From Concept to Implementation”, *IEEE Wireless Communications*, vol.20, no.2, pp.12-19, April 2013.

[16] J. Rodrigues, O.Pereira, and P. Neves, “Biofeedback Data Visualization for Body Sensor Networks”, *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 151-158, January 2011.

[17] L. Oliveira, A. Sousa, and J. Rodrigues, “Routing and Mobility Approaches in IPv6 over LoWPAN Mesh Networks”, *International Journal of Communication Systems*, vol. 24, no. 11, pp. 1445-1466, November 2011.

Authors

Mengting Zheng received her Ph.D. degree from Peking University, China in 2007. She was a visiting researcher in UCLA, USA from 2009 to 2014. Now she is an associate professor in AHUT, China. Her research interests include multimedia network and signal processing.

Wei Wang is now an associate professor in AHUT, China. His research interests include wireless communications and signal processing.