2-бөлім	Раздел 2	Section 2
Информатика	Информатика	Computer Science

IRSTI 20.23.17

Dynamic request distribution for enhanced Quality of Service

Joldasbayev S.K., Al-Farabi Kazakh National University,
Almaty, Kazakhstan, E-mail: serykjoldasbaev@mail.ru
Balakayeva G.T., Al-Farabi Kazakh National University,
Almaty, Kazakhstan, E-mail: gulnardtsa@gmail.com
Aidarov K.A., Al-Farabi Kazakh National University,
Almaty, Kazakhstan, +77013460160, E-mail: kanataidarov@yahoo.com
Chris Phillips, Newcastle University, Newcastle, Great Britain,
0191 208 7972, E-mail: chris.phillips@ncl.ac.uk

The emergence of Web-services as an open components supporting flexible and inexpensive set of distributed applications as well as exploiting them as a promising solution for integration with other application and providers of software-hardware resources is very much in demand. Using Web-services simplifies and enhances system functionality because of the availability of programs interaction with each other through Internet using open protocols. Thereby it is necessary to provide QoS provision issues such as distribution of request streams, enhanced efficiency of operations at peak loads. One of the ways to tackle these issues is to apply dynamic request distribution which ensures continuity of data transfer and processing ignoring crashes and equipment failures, redistribution of data flow to the available nodes in case of their loss. Abovementioned can be achieved using certain heuristics based on methods of Mathematical statistics and Probability modeling. Moreover, we provide analysis of methods and models developed for QoS provision.

Key words: Quality of Service, QoS, dynamic balancing, resource allocation, load balancing, queuing.

Қызмет көрсету тиімділігін арттыру үшін сұраныстарды үлестірудің динамикалық моделі

Жолдасбаев С.Қ., әл-Фараби атындағы Қазақ ұлттық университеті, Алматы қ.,
Қазақстан Республикасы, E-mail: serykjoldasbaev@mail.ru
Балақаева Г.Т., әл-Фараби атындағы Қазақ ұлттық университеті, Алматы қ.,
Қазақстан Республикасы, E-mail: gulnardtsa@gmail.com
Айдаров Қ.А., әл-Фараби атындағы Қазақ ұлттық университеті, Алматы қ.,
Қазақстан Республикасы, E-mail: kanataidarov@yahoo.com
Chris Phillips, Newcastle University,
Newcastle, Great Britain, E-mail: chris.phillips@ncl.ac.uk

Веб-сервистердің пайда болғаны икемді және қымбат емес кеңейтілген қосымшаларды қолдайтын ашық компоненттер, сондай-ақ оларды басқа қолданбалармен және бағдарламалық және аппараттық ресурстармен жабдықтаушылармен интеграциялау үшін перспективалық шешім ретінде пайдалану үлкен сұранысқа ие. Веб-қызметтерді пайдалану ашық хаттамаларды қолданып Интернет арқылы бір-бірімен бағдарламалық өзара әрекеттесудің болуына байланысты жүйенің жұмысын жеңілдетеді және жетілдіреді. Осылайша, QoS қамтамасыз ету мәселелерін шешу қажет, мысалы, сұраныс ағындарын бөлу, ең жоғары жүктемелерде операциялардың тиімділігін арттыру. Бұл мәселелерді шешу жолдарының бірі деректерді беру және өңдеудің үзіліссіздігін қамтамасыз ететін динамикалық сұранысты таратуды, сәтсіздіктер мен жабдықардың істен шығуын елемеуді, жоғалған жағдайда қол жетімді түйіндерге деректер ағынының қайта бөлінуін қамтамасыз ету болып табылады. Жоғарыда көрсетілген математикалық статистика мен ықтималдық модельдеу әдістеріне негізделген белгілі эвристиканы қолдана отырып қол жеткізуге болады.

Бұдан басқа, біз QoS-ті қамтамасыз ету үшін әзірленген әдістер мен модельдерді талдауды ұсынамыз.

Түйін сөздер: Қызмет көрсету сапасы, QoS, динамикалық теңдестіру, ресурстарды үлестіру, жүктемені теңдестіру, тапсырыс.

Динамическое распределение запросов для повышения качества обслуживания
Джолдасбаев С.К., Казахский национальный университет имени аль-Фараби,
Алматы, Республика Казахстан, E-mail: serykjoldasbaev@mail.ru
Балакаева Г.Т., Казахский национальный университет имени аль-Фараби,
Алматы, Республика Казахстан, E-mail: gulnardtsa@gmail.com
Айдаров К.А., Казахский национальный университет имени аль-Фараби,
Алматы, Республика Казахстан, E-mail: kanataidarov@yahoo.com
Крис Филлипс, Университет Нюкасл, Нюкасл, Великобритания,
E-mail: chris.phillips@ncl.ac.uk

Появление Web-сервисов как открытых компонентов, поддерживающих гибкий и недорогой набор распределенных приложений, а также их использование в качестве перспективного решения для интеграции с другими приложениями и поставщиками программно-аппаратных ресурсов, очень востребовано. Использование Web-сервисов упрощает и улучшает функциональность системы из-за доступности взаимодействия программ друг с другом через Интернет с использованием открытых протоколов. Таким образом, необходимо решать вопросы обеспечения QoS, такие как распределение потоков запросов, повышение эффективности операций при пиковых нагрузках. Одним из способов решения этих проблем является применение динамического распределения запросов, которое обеспечивает непрерывность передачи и обработки данных, игнорируя сбои и выхода из строя оборудования, перераспределение потока данных к доступным узлам в случае их потери. Вышеупомянутые могут быть достигнуты с использованием определенных эвристик на основе методов математической статистики и моделирования вероятностей. Кроме того, мы предлагаем анализ методов и моделей, разработанных для обеспечения QoS.

Ключевые слова: Качество обслуживания, QoS, динамическая балансировка, распределение ресурсов, балансировка нагрузки, очередность.

1 Introduction

In modern telecommunication networks, the character and volume of traffic transmitted and, accordingly, the provided info communication services are constantly changing. The increasing dissemination of web services, whether as an internal organizational infrastructure, or as part of paid services provided by the organization, issues related to service quality become very relevant and will ultimately be a significant factor in the success of service providers. For example, when customers contact you through the customer service centre's of services, they usually interact with automated support systems and expect an immediate answer. However, if the issue requires human intervention, it is likely that consumers will hold on the line, waiting for a response operator. Similarly, when the stock market is unstable and a large number of brokers and traders is online, as a rule, the trading platform can become overloaded, inhibiting or causing significant difficulties in the operation of such systems. The inability to buy and sell in a timely manner can cause serious financial losses.

2 Review of literature

To date, a large number of well-known scientists, for example, Kleinrock, S. Blake, D. Grossman, Z. Wang, Steklov VK, Berkman LN, as well as research centers such as Mobile Ad-hoc

Networks, Internet Engineering Task Force, Center for Embedded Networked Sensing deals with the management and distribution of traffic. However, despite the huge number of publications and the efforts of manufacturers, the task of constructing traffic patterns that best reflects its functioning in real conditions is still not solved [20-25].

Specifically, because reinforcement learning offers the potential to develop optimal allocation policies without explicit model knowledge by learning from the consequences of each action, existing works on ML algorithms mainly focus on reinforcement learning [11], [12]. They require neither an explicit system model nor an explicit traffic model to learn.

RL refers to a learning process, where a learning agent can learn to make appropriate decisions through interactions with an external environment [13]. Specifically, beyond the learning agent and the environment, a reinforcement learning system consists of a policy, a reward function and a value function. Let S be the set of environment states and A be the set of actions, respectively.

Another popular machine learning algorithm is the support vector machine (SVM). It has been widely applied for different areas such as pattern recognition, classification and data mining. However, SVMs are not preferred in on-line applications since the training and testing complexity of standard SVM are $O(nm + m^3)$ and (m) respectively, where n is the data size and m denotes the number of support vectors. On the other hand, some approximated methods have been proposed to reduce the complexity [14]. For example, [15] reduces the complexity to $O(nd_{2\ max})$, where d_{max} is the number of basis functions elected.

A few of works on machine learning algorithms have been proposed for the resource management problem [16-19]. For admission control, [8] derived a complex rule set that can be used to identify the optimal configuration for unobserved workload based on machine learning algorithms. [19] applied RL to configure parameters automatically in multi-tier Web systems, where eight parameters at web tier and application tier are selected to consist of the state space. For each parameter, there are three possible actions: increase, decrease and keep. The policy is based on the e-greedy method. In order to suppress the poor performance due to bad initialization, they proposed an algorithm to construct different initialization policies for different scenarios. For VM scaling, [16] proposed an iterative model training technique based on artificial neural network (ANN) to predict computing resource demand in virtual environments. [17] applied RL to train nonlinear approximators (e.g., multi-layer perceptrons) instead of the lookup table for VM horizontal scaling, where the state is defined as the request arrival rate and the action is to determine the number of servers allocated. Since the state space grows exponentially with the number of parameters in practice, the authors applied a nonlinear function approximator as an external policy to avoid poor performance that would be expected during online learning.

Recently, a few of works on fuzzy control for the resource management have been proposed in [10-12]. In [10], the admission control is conducted by fuzzy control in order to manage the QoS, where the turning parameter Maxclients in each interval is controlled by the fuzzy controller. For VM scaling, [11] attempted to capture the non-linear behaviors in VM resource usages by designing a fuzzy model estimator. The approache is divided into two steps. First, a fuzzy logic based modeling method is used learn the system behaviors without requiring any priori knowledge. Then a predictive controller predicts the resource demand of all VMs and takes actions based on this model. [12] proposed a neural fuzzy controller for percentile-based end-to-end delay guarantee through a virtualized multi-tier server cluster, where Gaussian

membership functions are first used to fuzzify the average service time, s i , and the variance of service time, $sigma\ 2_i$, distribution of requests at tier i, respectively. Then a fuzzy neural network is applied for online learning at the Inference stage. In addition, an output scaling factor is introduced to further enhance the performance. It is model-independent and capable of adapting control parameters through fast online learning. Compared with other supervised machine learning techniques, it does not require off-line training.

The urgency of the work on the material implementation of QoS in the Internet is a time-consuming task, but its solution will certainly affect the balancing of the load as a whole.

The quality of traffic management and distribution is characterized by such a notion as Quality of Service (QoS), which is the ability of a computer network to provide the required level of service to a given traffic within a certain technological framework [1].

The necessary service is characterized by many parameters, the main ones being the following:

– Bandwidth (bandwidth);
– priority;
– Delay of data transmission (delay);
– delay variation – jitter;
– loss of packets in the transmission of network data (packet losses).

In order to determine which simulation is most suitable and effective for managing and distributing traffic, typical problems that arise with bit transfer during periods of network congestion should be considered.

Developments in this area contribute to the effective operation of the IT service delivery system. Today there are huge resources for data processing, however, its effective application will contribute to the correct allocation of time and memory of the system, which are important factors in modern conditions. In addition, the information in the servers of the system, as well as requests for them, are steadily growing, which gives the demand for research in this direction relevant [2].

QoS (quality of service) – the ability of a network to provide the necessary service to a given traffic within a certain technological framework. "In this technical sense, this term means a set of methods for managing packet network resources. The quality of the connection is determined by four parameters:

1) Bitrate, describes the nominal bandwidth of the media. Depends on the bandwidth of the communication channel (Hz) and the signal-to-noise ratio;

2) Delay in the transmission of a packet (Delay), measured in milliseconds;

3) Fluctuation (jitter) of packet delay;

4) Packet loss. Determines the number of packets lost on the network during transmission [3].

To date, in modern telecommunications, one of the most urgent tasks is the transfer of traffic in compliance with a number of requirements for quality of service. Inefficient use of network resources, a large number of subscribers, as well as stringent requirements to the quality of service parameters may be the reason for the decline in the quality of services provided in multiservice data networks. Service quality technologies enable you to meet the workload or application maintenance requirements by measuring network bandwidth, monitoring network state changes (eg, used or available bandwidth), and prioritizing (or regulating) network

traffic [4]. Quality of service QoS refers to non-functional properties of interaction with Web services. These properties determine the degree of satisfaction in the query. QoS on customer terms can be grouped into three categories: performance, security and availability. These categories contain a set of various non-functional properties and metrics that determine the non-functional properties of the web service in quantitative forms in a certain order, linking it to unambiguous semantics. When measuring metrics, you should take into account the measurement indicators, facilities, organizations and the point in the network under which measurements are to be taken.

If several queue control algorithms are planned for wireless network design, for example, FIFO algorithm, priority maintenance algorithms, weighted queue algorithm, then in this case it is suggested to use the following order for comparison.

1. Calculation of the generation period

$$\tau = \frac{1}{f},$$

where τ is the generation period; f is the network frequency.

2. Determine the network speed

$$b = \frac{1}{m},$$

where b is the average service time; m is the data rate.

3. Bandwidth of channels

$$C = \sum_{k=1}^{K} \frac{1}{t_k}$$

where C is the throughput; t is the delay time; k is the number of channels.

Comparative analysis using the presented formulas will allow to judge, firstly, what equipment is necessary in this situation when building a computer network, which minimizes the delay in information transfer, and also effectively manages and distributes traffic. Secondly, what algorithm is better to use in that or other equipment.

In order to improve the quality of service, modern systems use intelligent service delivery systems that contribute to the improvement of the service delivery system.

For example, in [5-6] some solutions to the problems of availability and fault tolerance with the use of Apache Zoo Keeper technologies are proposed. The authors of this paper to solve the problem of distribution of flows, especially at high workloads, mainly, the transfer of some of the main loads to other servers (followers). of distribution of flows, especially at high workloads, mainly, the transfer of some of the main loads to other servers (followers). The proposed method is aimed at scenarios in which the Zab protocol is potentially a bottleneck in performance, this can happen for some reasons: (1) the intensity of work with intensive recording; (2) the number of clients is quite large; or (3) a large number of ensemble N, $N > 3$. In other words, the method reduces the traffic of messages, both incoming and outgoing, from the leader. This is achieved by changing the behavior of followers in two simple but important ways, while at the same time supporting the structure of Zab itself, which is well understood and implanting the structure. Thus, the costs associated with reaching an agreement were reduced, lower overheads and increased productivity.

3 Materials and methods

The urgency of the work on the material implementation of QoS in the Internet is a time-consuming task, but its solution will certainly affect the balancing of the load as a whole, both globally.

To describe the QoS work in the physical representation, that is, the definition of what it is, let's go down to the hardware implementation. It will be appropriate to determine that in the work we do not touch this level (the hardware part) and no change in its architecture or the like. solutions is not a task. At the same time, without describing the hardware structure, it will be difficult to state the task at the middleware & software level.

Implementation of dynamic QoS solutions for servers of large-scale clusters exists relatively long ago. But the use of the Internet of things in different approaches can bear other promising solutions. The use of fairly simple elements of the Internet things from improvised tools in turn can have a grandiose significance for QoS in general.

There are three models of implementing QoS: the best possible; integral and differentiated. In the hardware, we can define this as the following:

The best possible type of service is realized on the network, when everything is done to deliver the package, but nothing is guaranteed (for example, FTP or HTTP). The integrated type of services (RFC-1633, 1994) was developed first and implemented by point-to-point redundancy (RSVP protocol, 1997, section 4.4.9.6). RSVP provides a signaling mechanism for configuring remote routers to obtain the desired QoS [3], [8]. The protocol is oriented to work with three types of traffic: best efforts (normal transmission of IP-data without connection), sensitive to the transmission speed and sensitive to delays. Traffic sensitive to download requires the formation of a channel with guaranteed bandwidth. The application is forced to put up with certain delivery delays (a class of services with a guaranteed bit rate in seconds).
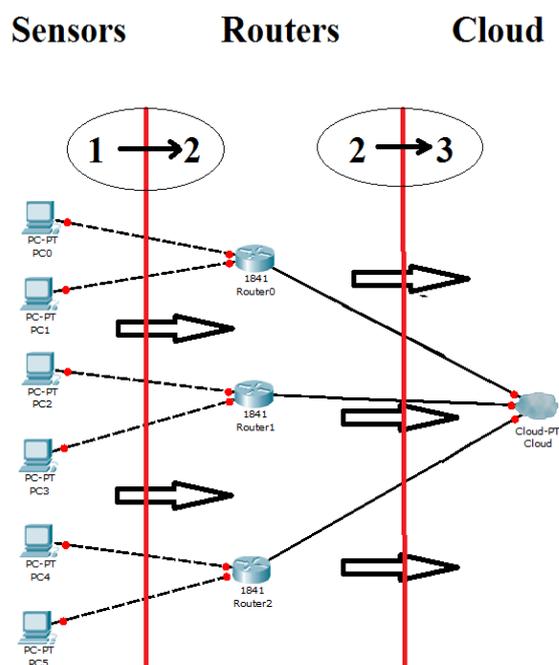

Figure 1: The general picture of network connection for dynamic model

3.1 Mathematical model

The third type of traffic (sensitive to delays) guarantees a minimum delay and a low variance in the delivery time. The capacity can vary. An example of this type of traffic can serve as voice or video. RSVP defines two types of services for this type of traffic: a service with controlled delays and a predictable service (for real-time applications (video conferences and telephone conversations) [6].

It is known that the transmission of information through Internet services occurs in a portioned, in the form of a session, containing many individual requests. These requests are expressed as a queue of atomic units that have a finite number of elements, called threads. Each request in the stream is a package, each of which has an application and a limited number of m. Therefore, one packet inside the stream at a certain point in time will be processed in only one of the limited sets of available N processors [4].

Streams can be m different types with different performance requirements. a stream of a i-type contains $k(i)$ applications sent at the frequency of $l(i)$ applications per second. These values will be needed later in the description of various QoS heuristics.

3.2 Development of algorithm of performance evaluation

Suppose that for each i-th stream, K_i - the total number of packets organized in the FIFO queue. The flow has important characteristic, λ_i is the rate of arrival of the i-th stream. Each processor has its own characteristic b_i - the average processing time, which means the average time of finding the packets in the processing processor. And each from this processor can process only one packet at a time. Each packet in the stream has a characteristic L_j^i - delay j -th packet in i -th stream. Delay means the total time that a package has been in the system, from the moment it hits the queue, until it leaves the processor after processing. On the basis of this we introduce the following variable: W_j^i is the waiting time of the packet and we define it as

$$W_j^i = L_j^i - b_i$$

where $j = 1, 2, \ldots, K_i$ and i stream index. We introduce one more variable Q_i - the average waiting time for the i-th stream and express it as follows:

$$Q_i = \sum_{j=1}^{K_i} W_j^i$$

The application vendor leases server resources from the cloud and provides user work execution services, which is expressed in the form of a Service Level Agreement (SLA). This agreement contains a list of quality parameters for servicing methods and means of their control, the time of the supplier's response to the consumer's request, as well as penalties for violations of this agreement. It consists of three components:

1. A description of the quality of service guarantee that the cloud platform provides to the user;

2. A revenue-generating scheme that is used by the platform for charging for the services provided;

3. Penalties for violations of the agreed performance guarantee under the contract.

SLA in our case is defined as follows:

$$Q_i \leq \sigma_i, i = 1, 2, \ldots, M$$

Here M is the total number of streams accepted for maintenance, σ_i is the profit received for completing the maintenance of the i-th stream (processing of all packets of this stream). P_i is penalties for violations of the SLA.

We define a new set of variables:

$$profit_i = \sigma_i - A_i \times cost \times duration_i$$

Which A_i - number of dedicated processors for the i-th stream, $cost$ - a fixed number of units per unit time, $duration_i$ - amount of time spent processing.

To determine system performance, we use the utility function. R - the average profit received by the service provider per unit time is the main metric considered in this work.

$$R = \sum_{i=1}^{m} a_i[c_i - r_i P(W_i > q_i)]$$

where a(i) is the average number of i-type flows received per system per unit time, $P(W_i > q_i)$ is the probability of exceeding the observed waiting time for streams commitment q_i.

Heuristics with the same flows can be optimized by applying one or more steps to improve the heuristic, similar to another. The ultimate goal of resource allocation and heuristics of admission control is to maximize this value (R).

4 Results and discussion

To improve the efficiency of providing services at the software and middleware levels, it is proposed to use the dynamic model of an intelligent system at high loads without affecting the hardware of the system as a whole. The method of solution and the mathematical model of regulation of load balancing are presented in the article according to the statement of the problem.

The results of this work at this stage can be defined as the proposed effective method of solution, as well as the correct wording of the problem itself.

An algorithm for the functioning of a new model for carrying out extensive computational experiments is being developed.

In the future, emphasis will be placed on numerical solutions and simplifying the problem to the required level.

5 Conclusion

In a dynamic system, instead of the limited resources N, a non-constant number of processors with the total amount limitation will be taken into account: $I \in N$.

In this connection we use a dynamic solution to the problem of distribution of queries to solve the problems of distribution of flows, in order to increase the efficiency of operations, especially at high loads, which will ensure the continuity of data transmission and processing, ignoring equipment breakdowns and outages, distributing data on accessible nodes, in case of their loss.

References

[1] Hyunyoung Kil, Reeseo Cha, Wonhong Nam. Transaction history-based web service composition for uncertain QoS. *International journal of web and grid services*, vol. 12 (2016): 42.

[2] Balakayeva G.T., Aidarov K.A. Research of algorithms and methods of load balancing and construction of models for queuing networks. *Proceedings of the International Conference on Computational and Applied Mathematics "VPM'17"in the framework of the Marchuk Scientific Readings, Novosibirsk, June 25 – July 14 [Electron. resource]*, (2017): 17–21.

[3] Goldstein B.S., Marshak MA, Mishin E.D., Sokolov N.A., Tum A.V. "Indicators of the functioning of the multiservice communication network of public use". *Journal of Communication Engineering*, no. 3–4 (2009): 17.

[4] Balakayeva G., Aidarov K., Simulation of load balancing algorithms based on queuing networks. *Abstracts of VI Congress of the Turkic World Mathematical Society (TWMS 2017), Astana*, (2017): 313.

[5] EL-Sanosi I. and Ezhilchelvan, P. Improving zookeeper atomic broadcast performance by coin tossing. *In European Workshop on Performance Engineering, Springer*, (2017): 249–265.

[6] Flannagen E. Michael. Syngress (2001) "Administering CISCO QoS in IP-Networks", *Syngress Media, ISBN 1928994210, 9781928994213*, (2001): 519.

[7] Goldstein B.S., Marshak M.A., Mishin E.D., Sokolov N.A., Tum A.V. Kontrol pokazatelei kachestva obsluzhivanya s uchotom perekhoda k seti svyazi sleduyushego pokolenya [Control of quality of service indicators, taking into account the transition to a next-generation communication network]. Tekhnika Svyazi, no 1 (2009).

[8] Andrzejak A., Arlitt M., Roila J. Bounding the Resource Savings of Utility Computing Models. *Technical Report HPL-2002, Internet Systems and Storage Laboratory, HP Laboratories*, (December 2002): 339.

[9] Kharchenko V, Illiashenko O, Boyarchuk A, Sklyar V, Phillips C Emerging curriculum for industry and human applications in Internet of Things. *In: 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) Bucharest, Romania: Institute of Electrical and Electronics Engineers Inc.*, (2017): 918-922.

[10] Chase, J. S., Anderson, D. C., Thakar, P. N., Vahdat, A. M., Doyle, R. P. Managing energy and server resources in hosting centers. *ACM SIGOPS Operating Systems Review.*, no 35(5), (2001): 103.

[11] Lee, Y. C., Zomaya, A. Y. . Energy efficient utilization of resources in cloud computing systems. *The Journal of Supercomputing*, 60(2), (2010): 268–280.

[12] Enokido, T., Aikebaier, A., Takizawa, M. A Model for Reducing Power Consumption in Peer-to-Peer Systems. *IEEE Systems Journal*, 4(2),(2010): 221–229.

[13] Liu, S., Ren, S., Quan, G., Zhao, M., Ren, S. Profit Aware Load Balancing for Distributed Cloud Data Centers. *2013 IEEE 27th International Symposium on Parallel and Distributed Processing*, (2013): 611–622.

[14] Vakilinia, S., Heidarpour, B., Cheriet, M. Energy Efficient Resource Allocation in Cloud Computing Environments. *IEEE Access*, 4, (2016), 8544–8557.

[15] Zhang, W., Zhang, Z., Chao, H.-C. . Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management. *IEEE Communications Magazine*, 55(12),(2017): 60–67.

[16] Nagpure, M. B., Dahiwale, P., Marbate, P. An efficient dynamic resource allocation strategy for VM environment in cloud. *2015 International Conference on Pervasive Computing (ICPC)* (2015).

[17] Mukherjee, M., Shu, L., Wang, D. Survey of Fog Computing: Fundamental, Network Applications, and Research Challenges. *IEEE Communications Surveys and Tutorials*, (2018): 1–1.

[18] Bodenstein, C., Schryen, G., Neumann, D. Energy-aware workload management models for operation cost reduction in data centers. *European Journal of Operational Research*, 222(1), (2012): 157–167.

[19] Mohammad Ali, H. M., El-Gorashi, T. E. H., Lawey, A. Q., Elmirghani, J. M. H. Future Energy Efficient Data Centers With Disaggregated Servers. *Journal of Lightwave Technology*, 35(24), (2017): 5361–5380.

[20] Hameed, A., Khoshkbarforoushha, A., Ranjan, R., Jayaraman, P. P., Kolodziej, J., Balaji, P., . . . Zomaya, A. A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems. *Computing*, 98(7), (2014): 751–774.

[21] Ge, Y., Zhang, Y., Qiu, Q., Lu, Y.-H. A game theoretic resource allocation for overall energy minimization in mobile cloud computing system. *Proceedings of the 2012 ACM/IEEE International Symposium on Low Power Electronics and Design - ISLPED '12.*, (2012): 279-284.

[22] Kliazovich, D., Arzo, S. T., Granelli, F., Bouvry, P., Khan, S. U. e-STAB: Energy-Efficient Scheduling for Cloud Computing Applications with Traffic Load Balancing. *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, (2013):7-13.

[23] Aikebaier, A., Yang, Y., Enokido, T., Takizawa, M. Energy-Efficient Computation Models for Distributed Systems. *2009 International Conference on Network-Based Information Systems* (2009): 424-431.

[24] Sharma, B., Wood, T., Das, C. R. HybridMR: A Hierarchical MapReduce Scheduler for Hybrid Data Centers. *2013 IEEE 33rd International Conference on Distributed Computing Systems* (2013): 102-111.

[25] Gao, Y., Guan, H., Qi, Z., Song, T., Huan, F., Liu, L. Service level agreement based energy-efficient resource management in cloud data centers. *Computers and Electrical Engineering*, 40(5), (2014): 1621–1633.