Heterogeneous Cloud Systems and Criteria for Enhanced Performance

Jarmila Škrinárová
Department of Computer Science
Matej Bel University
Banská Bystrica, Slovakia
jarmila.skrinarova@umb.sk

***Abstract*—Cloud systems are managed on the basis of autonomous systems. We present criteria that are suitable for optimization or improvement of modern cloud systems. In every operating system, task scheduling is very important. In clouds systems, where large amount of tasks runs on numerous machines, optimized task scheduling leads to significant reduction of computing time. Cloud providers have to comply with service level agreement from technical and from the quality of service point of view. For this reason, we specify quality of service criteria and limits for service level agreement violation. Clouds are virtualized by virtual machines or containers. We show approaches for power consumption minimization. Fog computing helps to improve middleware technology between cloud computing and the IoT devices. We specify criteria for correct decomposition of parallel and distributed application. We conclude that understanding of effective work of the system, can improve design and implementation of parallel and distributed application.**

***Index Terms*—cloud systems, heterogeneous clouds, microclouds, microservices, containers, criteria of quality**

I. Introduction

In this lecture, we try to draw attention to the diversity of cloud systems. Users usually perceive clouds as storage for their files, or platform for execution of their application. Cloud systems are successfully used for large-scale and time-consuming scientific computations, for which high hardware requirements are characteristic (expressed by the number of $CPUs$, $GPGPUs$, memory capacity, connection speed, bandwidth, and so on) [1].

Therefore, some clouds are centralized by default and contain a large number of computing nodes and are often located in data centers with large data storages [32]. They are equipped with management systems for scheduling of tasks on computing machines. Modern management systems monitor and predict system load and a number of other parameters in order to achieve optimized system behavior (minimum task computation time, maximum system throughput, minimum electricity consumption, and so on) [3].

This also results in a number of criteria that clouds have to meet. From the point of view of the system, the task that comes into the system must be available as soon as possible. To meet this requirement, the deployment of containers instead of virtual machines is preferred in current systems. In addition, modern approaches try to minimize the availability time (start time) of the container. It is also important that the task, which runs in the system, has shortest possible execution time. For this purpose, the optimization criterion: ”completion time of the last task”, or $makespan$ is traditionally and successfully used. This is a basic criterion that is suitable for all cloud systems [1], [2].

Currently, the scientific activity related to cloud systems is focused on the use of clouds for processing data related to the Internet of Things (IoT).

Clouds for IoT can be implemented as microclouds and are placed on the edge of the network, as close as possible to the place where the data is generated [5], [25].

It is also important to minimize the consumption of electric energy. Minimization, or reducuction of electricity consumption is essential in all types of cloud systems. Some approaches are used to optimize consumption in centralized clouds, and other approaches are needed to reduce consumption for processing a large number of tasks on a relatively large set of microclouds.

The method of consolidating virtual machines or containers is successfully used to reduce electricity consumption. The modern approach to reducing electricity consumption is based on brownout technology.

All heterogeneous cloud systems have to work in such a way that the quality conditions defined in the SLA (in the agreement between the client and the service provider) are met [3], [4]. Some are defined by the start time of computation or completion time of the task, bandwidth or system throughput. Therefore, it is often necessary to solve optimization criteria from the system's point of view in combination with SLA criteria. Some approaches define the so-called SLA violations as a limit or critical points that cannot be exceeded because it would be impossible to meet the SLA.

At the end of the lecture, we specify criteria for correct decomposition of parallel and distributed application. Because, understanding of effective work of the system, can improve design and implementation of parallel and distributed application.

II. Criteria Suitable for Improvement of Modern Cloud Systems

Task scheduling and completion time of last task in batch of tasks minimization. In computing systems, the term task scheduling denotes allocation of tasks to computational resources for the specified time interval. Cloud systems for high performance computing contain a large number of machines and are processing a huge number of large – scale data. Scheduling problem is defined by the set of machines, the set of tasks and optimization criterion. Since creation of schedule is $NP\text{–}complete$ problem, we need to use an optimization method [6], [7]. Typical and very useful criterion is completion time of last task in batch of tasks C_{max}. We are searching for schedule with the smallest C_{max}. In the last years, number of approaches with effective and evaluated implementations, that quickly and pseudo optimally scheduled tasks in cloud systems was introduced. We also used various approaches to this problem in our works [28], [29]. For this purpose, authors of [16], [17], [30] created the cloud scheduling simulators.

Criteria for quality of service. Service level agreements between clients and cloud providers also include the criteria for quality of service. There are various cloud systems. For example, clouds for high performance computing, clouds for Internet of Things (with large amount of devices), microclouds and others. Modern technologies have various functionalities and they can have different requirements on quality of service. The cloud systems are centralized and IoT devices are allocated on the edge of the network. Fog computing represents middleware technology that is situated between cloud system and IoT devices. The typical requirements for a cloud system are related to the quality of service, such as: reservation of computing resources in advance, the last time for starting the specified task, deadline for finishing the task, limit for maximal slowdown of task. On the other hand, for IoT connection with a cloud another requirements are specified: limits for response time, bandwidth, latency, uninterruption and reliability in data stream transportation from large amount of devices [26], [27].

Criteria for power consumption minimization. Cloud system can be virtualized by virtual machines or containers. Basic criterion is to minimize the number of active servers, if the system is underloaded. An effective method for clouds with virtual machines is consolidation of the virtual machines. This idea is based on migration of all virtual machines from underloaded servers which allows us to switch these idle servers to low-power mode [14], [15], [20]. Second optimization criterion is to minimize amount of virtual machines which were switching on, after switching on the same virtual machine in specified time interval [9 - 13]. For clouds that are virtualized by containers [18], [19] brownout technology can be used as a method for reducing power consumption. In this method we can set the threshold for overloaded servers. If there are overloaded servers detected, probability of triggering a brownout and utility reduction for every overloaded server is computed. The lowest utilized containers are deactivated [8], [21], [23].

Limits for Service Level Agreement violation. For autonomous management of cloud system it is necessary to have implemented such algorithms, that are optimized by particular previous criteria and at the same time algorithms that arranged that limits for Service Level Agreement violation will not be exceeded [18].

For example, in microservices [22] - typical limit for Service Level Agreement violation is ratio between number of requirements without responses (untill some time interval) and all requirements. For other type of services, we can compute limit for Service Level Agreement violations time per one host ($SLATH$) as ratio between total time when host was fully utilized (leading to $SLAV$) and total time when host was active. For all hosts in the cloud system this limit ($SLATAH$) can be computed as an average of Limit for Service Level Agreement violations time of active hosts.

Container start time minimizing. Criteria for fog computing can be various. One of the important criteria in this area is to minimize start time of container. Standard container is managed by container engine, e.g. Docker. For acceleration of the start time of container it is possible to modify docker file system. It is necessary to identify a set of files that are needed for starting of the container and entire directory structure of the original file system image. Before starting of the container is finished so called FogDocker starts necessary files, mounts virtual file system and creates directory tree. FogDocker significantly reduces starting time of a container [24].

Correct decomposition of parallel and distributed application. We introduced various criteria which are important for effective work of the cloud system. Understanding of the criteria can improve design and implementation of parallel and distributed application. In process of designing parallel and distributed applications which are suitable for cloud computing it is necessary to respect some criteria. Suitably designed application helps to loadbalance the system, to minimize communication between tasks and to minimize waiting in the system. There are two main and very effective criteria that we evaluated:

1) to search such decomposition of task that computation times for all subtasks will be identical.
2) to minimize computation time for every subtask and at the same time for whole application [31].

Decomposition of the computational task is sometimes more difficult. For example, if each subtask contains recursions with different numbers of recursion calls, then computation times of these subtasks are very diverse. This causes a high load and time imbalance. With the use of classification it is possible to compute all subtasks with similar or same numbers of recursion calls (are in the same class) in close to identical times [32].

Acknowledgment

Computing was performed in the High Performance Computing Center of the Matej Bel University in Banska Bystrica using the HPC infrastructure acquired in project ITMS 26230120002 and 26210120002 (Slovak infrastructure for high-performance computing) supported by the Research & Development Operational Programme funded by the ERDF.

The research was partially supported by the grant of The Ministry of Education, Science, Research and Sport of Slovak Republic - Implementation of new trends in computer science to teaching of algorithmic thinking and programming in Informatics for secondary education, project number KEGA 018UMB-4/2020.

The support of Advtech_AirPollution project (Applying some advanced technologies in teaching and research, in relation to air pollution, 2021-1-RO01-KA220-HED-000030286) funded by European Union within the framework of Erasmus+ Program is gratefully acknowledged.

References

[1] J. Skrinarova: Elastic cluster. (In Slovak) Elastický klaster. Banská Bystrica: 2017, Belianum ISBN 978-80-557-0642-9

[2] S. Palúch, S. Peško: Quantitave methods in logistics (In Slovak) Kvantitatívne metódy v logistike. ŽU Žilina. 2006. ISBN-80-8070-636-0, 185p.

[3] R. Buyya, J. Broberg, A. Goscinski: Cloud computing, principles and paradigms. New Yersey: 2011, John Wiley & Sons ISBN 978-0-470-88799-8

[4] P. Mell, T. Grance: The nist definition of cloud computing recommendations of the national institute of standards and technology. Tech. rep., NIST - U.S. National Institute for Standards and Technology, 2011.

[5] Next generation cloud computing: New trends and research directions. https://doi.org/10.1016/j.future.2017.09.020

[6] J. Yan, Y. Huang, A. Gupta, A. Gupta, C. Liu, J. Li, L. Cheng: Energy-aware systems for real-time job scheduling in cloud data centers: A deep reinforcement learning approach. Computers and Electrical Engineering, 99, (2022), 107688

[7] S. A. Murad, A. J. M. Muzahid, Z. R. M Azmi, M. I. Hoque, M. Kowsher. A review on job scheduling technique in cloud computing and priority rule based intelligent framework. Journal of King Saud University – Computer and Information Sciences.

[8] M. Xu, R. Buyya. BrownoutCon: A software system based on brownout and containers for energy-efficient cloud computing. The Journal of Systems and Software.

[9] U. Arshad, M. Aleem, G. Srivastava, J. C.-W. Lin: Utilizing power consumption and SLA violations using dynamic VM consolidation in cloud data centers. Renewable and Sustainable Energy Reviews

[10] A. Beloglazov, R. Buyya: Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers. Concurrency Computat.: Pract. Exper., 2012, 24: 1397-1420. https://doi.org/10.1002/cpe.1867

[11] N. B. Rizvandi, J.Taheri, A. Y. Zomaya. Some observations on optimal frequency selection in DVFS-based energy consumption minimization. Journal of Parallel and Distributed Computing, Volume 71, Issue 8, 2011, Pages 1154-1164, ISSN 0743-7315

[12] S. Ilager, K. Ramamohanarao, R. Buyya: ETAS: Energy and thermal-aware dynamic virtual machine consolidation in cloud data center with proactive hotspot mitigation. Concurr Comput: Pract Exper 2019;31:e5221.

[13] R. Zolfaghari, A.M. Rahmani: Virtual machine consolidation in cloud computing systems: Challenges and future trends. Wirel Pers Commun 2020;115:2289–326.

[14] C. Gu, H. Huang, X. Jia: Power metering for virtual machine in cloud computing-challenges and opportunities. IEEE Access 2014;2:1106–16.

[15] R. Buyya, A. Beloglazov, J. Abawajy: Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges. In: Parallel and distributed processing techniques and applications, Vol. 106. 2010, p. 116–24.

[16] R.N. Calheiros , R. Ranjan , A. Beloglazov, C.A. De Rose, R. Buyya: CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Softw - Pract Exp 2011;41:23–50.

[17] A. Hussain, M. Aleem, M.A. Iqbal, M.A. Islam: Investigation of cloud scheduling algorithms for resource utilization using cloudsim. Comput Inform 2019;38:525–54.

[18] Z. Zhou, J. Abawajy, M. Chowdhury, et al.: Minimizing SLA violation and power consumption in Cloud data centers using adaptive energy-aware algorithms. Future Generation Computer Systems 2018, 86, 836 – 850

[19] N. Gholipoura, E. Arianyanb, R, Buyya: A novel energy-aware resource management technique using joint VM and container consolidation approach for green computing in cloud data centers. Simulation Modelling Practice and Theory. 2020, 104, https://doi.org/10.1016/j.simpat.2020.102127

[20] K.H. Kim, A. Beloglazov, R. Buyya: Power-aware provisioning of virtual machines for real-time cloud services. Concurr. Comput. Pract. Exper. 2011, 23 (13), 1491–1505.

[21] C. Klein, M. Maggio, K.-E. Arzén, F. Hernández-Rodriguez: Brownout: build- ing more robust cloud applications. In: Proceedings of the 36th International Conference on Software Engineering, pp. 700–711. 2014.

[22] S. Newman: Building microservices. 2015.

[23] A.N. Toosi, C. Qu, M.D. de Assuncao, R. Buyya: Renewable-aware Geographical Load Balancing of Web Applications for Sustainable Data Centers. Journal of Network and Computer Applications (JNCA), Elsevier, 2017, 83, pp.155-168. DOI:10.1016/$j.jnca$.2017.01.036

[24] L. Civolani, G. Pierre, P. Bellavista: FogDocker: Start Container Now, Fetch Image Later. UCC 2019 - 12th IEEE/ACM International Conference on Utility and Cloud Computing, Dec 2019, Auckland, New Zealand. pp.51-59, DOI:10.1145/3344341.3368811.

[25] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, D.S. Nikolopoulos: Challenges and opportunities in edge computing, in: IEEE International Conference on Smart Cloud, 2016, pp. 20–26.

[26] R. Buyya, S.N. Srirama: Fog and Edge Computing: Principles and Paradigms. 2019, John Wiley & Sons, Inc. ISN 9781119524984

[27] T.-A. N. Abdali, R. Hassan, A. H. M. Aman, Q. N. Nguyen: Fog Computing Advancement: Concept, Architecture, Applications, Advantages, and Open Issues. IEEE Access. 2021. DOI: 10.1109/$ACCESS$.2021.3081770

[28] J. Skrinarova, L. Huraj, V. Siladi. A neural tree model for classication of computing grid resources using PSO tasks scheduling, NNW2013. DOI: 10.14311/NNW.2013.23.014

[29] J. Skrinarova: Implementation and evaluation of scheduling algorithm based on PSO HC for elastic cluster criteria. Open computer science, 2014.

[30] J. Skrinarova, M. Povinsky. Parallel simulation of tasks scheduling and scheduling criteria in high-performance computing systems. JIOS 2016.

[31] J. Skrinarova, A. Dudas. Optimization of the Functional Decomposition of Parallel and Distributed Computations in Graph Coloring With the Use of High-Performance Computing. IEEE Access 2022, DOI: 0.1109/$ACCESS$.2022.3162215

[32] M. Kvet, J. Papan: The Complexity of the Data Retrieval Process Using the Proposed Index Extension. (2022) IEEE Access, Vol.10, pp. 46187-46213. DOI: 10.1109/$ACCESS$.2022.3170711