ISSN: 2456 - 4664
International Journal of Advanced Trends in Engineering and Technology
Impact Factor 5.965, Special Issue, February 2018 – Conference Proceedings
National Conference on Recent Trends in Management Studies & Computer Applications
(KRUPACON 2017) On 21st & 22nd September 2017 Organized By
Krupanidhi Group of Institutions, Bangalore, Karnataka

EXECUTION TIME REDUCTION IN MOBILE CLOUD COMPUTING

Vijayamala S Yakri* & Anuradha N**
* Krupanidhi Group of Institutions, Bangalore, Karnataka
** Krupanidhi Degree College, Bangalore, Karnataka

Cite This Article: Vijayamala S Yakri & Anuradha N, "Execution Time Reduction in Mobile Cloud Computing", International Journal of Advanced Trends in Engineering and Technology, Special Issue, February, Page Number 13-15, 2018.

Abstract:

Mobile technology developments soon lead to numerous computer-intensive activities. However these activities pose difficulties to run on mobile devices due to resource constraints. A cloud migration was implemented to conduct a cloud work and then return the result to the consumer mobile device as a solution to this issue. In this text, an algorithm for cloud migration decisions was suggested in order to decide if execution on a cloud server is feasible rather than a mobile computer. In addition, there has been an inquiry into performances between mobile and cloud executions, showing that the completion period of a mission can be shortened by 6 to 8 times while a cloud server is being used.

Key Words: Compute-Intensive; Task Migration; Mobile Cloud; Decision Algorithm; Cloud Computing

Introduction:

Mobile devices have evolved in order to accommodate the many computer-intensive activities involved in image processing, multimedia, accessible computing, increased reality, and mobile health care, for example, mobile phones, laptops, wearable's, etc. [1]. [2] Mobile computing has developed mobile technology. In 2016, a new report found that there were 149.3 billion Smartphone app installs in the country, and in 2021 this is projected to amount to 352.9 billion. These applications produced $88.3 billion in revenue and by the end of 2020 were valued at about $188.9 billion [2]. While mobile devices have powerful hardware every day, they also do not have a huge amount of computing power and memory to cope with certain recent applications. Fortunately, mobile cloud infrastructure has been enforced to ease these limitations and support ever-growing demand for mobile computing and storage [3]. Mobile cloud computing requires computational programs or functions to be transferred to a central cloud server with massive processing and storage facilities. In general computing functions carry out more complicated calculations with a limited volume of data on a mobile computer and thus use more power [4]. In addition, several recent activities on mobile devices cannot be carried out because they cannot fulfill the required time constraint. Cloud-based execution assists with these tasks when performance time can be greatly decreased in contrast with the amount of data transmission. Consequently, cloud-based execution migration algorithms are constant for these functions. However the reliable estimation of migratory viability as mission migration brings on additional overheads, e.g. network connectivity and data transfer times, is a major problem in migration. This paper provides a description of the key contributions.

- A decision-making algorithm to assess the feasibility of migrating a mission from a mobile computer to a cloud service to minimize latency was suggested and introduced.
- The Amazon EC2 cloud server experimental test bed [5] for testing.

Literature Review:

Any work on task migration strategies has also been performed to decrease the success time of a task. In [6] Chun et al proposed to transfer and to perform part of a process from a remote server with the modular task partitioning scheme. It uses static analysis and dynamic profiling to reduce runtime and energy consumption for partitioning tasks. Kosta et al. suggested a platform called Think Air [7] to move mobile activities into the cloud using a method-level migration and virtualization technology of Smartphone. In order to run tasks in parallel and to achieve higher overall performance, the work employed elastic and flexible VM imaging. The results of an architecture that included two clone copies of the mission, namely Off-clone and Back-Clone, stored on a cloud server and used to migrate and save respectively, are analyzed by Barbera et al. in [8].

The percentage efficiency of a bandwidth and energy usage device was calculated through a real test bed with 11 Smartphone and Amazon EC2 public cloud instances. Kumar and collaborative authors summarized a review of the exciting computational migration technologies and architectural approaches [9] to take migration choices for saving energy and optimizing mobile device efficiency. In comparison, Khan et al. with their advantages and drawbacks in [10] outlined the latest mobile cloud computing architecture and task model. Cheng et al. suggested an interface for cloud migration and an algorithm in [11] to move certain computational activities from wearable systems to a local mobile computer or remote cloud so they can perform heavy computing tasks.

Migration of tasks involves estimation and a huge number of transfers of data. Computer-intensive operations in general require less data transfer compared to an immense amount of sophisticated processing. Furthermore, some megabits of mobile networks have already grown to support several Gigabits for Wi-Fi networks [12-16]. Khoda et al. recently revealed that migration from mobile devices from local to remote cloud will optimise energy savings and keep the consumer activities in 5G networks purely latentious. The key emphasis of this paper is a migration policy algorithm that performs a cloud role by comparing it with mobile device results [17-22].

Proposed Algorithm for Task Migration:

A mathematical formula for computational activity executions at a cloud platform is given in this section. It introduces the architecture to allow mobile device migration to a cloud server and provides an algorithm for determining the feasibility of migrating [23-24].

Architecture for Task Migration:

Calculation-intensive tasks involve several CPU cycles that makes it difficult to operate on mobile devices. This post looks at a two-layer mobile application architecture. Level 1, where it is suitable and level 2 has a greater computing capacity, is responsible for facilitating migration to a remote server. The cloud server offers a virtual machine instance that provides a certain amount of services to perform migrated tasks using the Service Level Agreement (SLA) for a mobile computer. Where a mobile computer has to perform a computing process, depending on the scale of the task, it calculates the time of the device and the time of server completion that cloud execution and contact time are needed. In order to approximate the contact time, a machine uses the database of past communications from the mobile device to the cloud server. 3G and Wi-Fi technologies are considered for networking and data transfer. Any other networking network infrastructure is nevertheless applicable. If the projected time for club fulfillment is shorter than the runtime for the mobile device, the mobile device pushes an issue into the cloud. As services become available, the cloud server completes the migrated task and returns data to the mobile device. In the case that the cloud service displays a mobile device error message and the job on the mobile device will be executed.

Experimental Setup and Results:

The experimental setup and findings are explained in greater detail in this section.

- Experimental Test Bed: An experimental test bed was developed based on the proposed algorithm to measure the efficiency changes in the cloud. The Bubble sorting task was used to model a computing task on a mobile computer. A collection of unsorted array components from the mobile device are moved to the cloud server and the sorted array through the contact network is returned to the mobile device. An Amazon Elastic Compute Cloud (EC2) instance was used to provide cloud service capabilities. The Wi-Fi and 3G applications of networking were used to migrate the computer-intensive cloud activities. Only mobile execution time is called the completion time for the execution of tasks on a mobile computer. Cloud completion period, on the other hand, requires time for measurements and contact. The emulation method uses a sequence of compute-intense tasks based on small array sizes and wide array size to observe the efficacy of the migration decision-making algorithms. Small size range between 1000 and 6000 (4 KB to 24 KB), and big sizes range between 10000 and 60000 (40 KB to 240 KB). The experimental findings for smaller displays help to define the stage between mobile execution and cloud migration, while large display sizes are used to identify the possible benefits in terms of cloud migration completion times. Both measurements were performed 20 times and the results obtained are summed. In determining the inclination and time variance in the execution of the assignment, confidence interval has been added.
- Impact on Small Array Sizes: To test the performance of the algorithm suggested in section III.B and results, first, between cloud complete with Wi-Fi connectivity and smart phone execution. The algorithm only performs the task for the array scale 1000 on mobile device. The role is migrated to the cloud server for array scale 2000 or greater. The judgment efficiency of the proposed 3G and mobile execution cloud algorithm is highlighted. It should be remembered that the algorithm conducts the smart phone sorting task to a scope of 4000 and transferred to the cloud for a bigger scope. The completion time of the assignment is seen to increase on a linear basis as the table size grows. Mobile execution can be conducted with array sizes up to 1000 regardless of the networking technology on a cloud-based basis. It should be remembered that cell execution choice for array sizes in 3G networks can be expanded to up to 4000. However the framework still favours cloud migration with 3G networks and array sizes 5000 or larger. The result shows that connectivity times are improved linearly with increasing the scale of the array for Wi-Fi and 3G. Wi-Fi is never simpler than 3G, which is also technically predicted. It was found that contact time increases because of the increased amount of data processing due to the increasing array size. In comparison, the lag of 3G connectivity is greater than WiFi. Although cloud deployment results way higher than the cut-off point (1000 and 4000 arrays) as seen in Figure 3. mobile application execution also profits. This is because the total connectivity benefits in migrated CCs on the cloud server are eliminated by a significant volume. However, cloud migration is often useful after the cutoff point as cloud completion time eliminates overhead connectivity for a higher result than cell device execution.
- Impact on Large Array Sizes: In the case of cloud, completion time marginally increases with the array size while completion time for mobile execution easily increases. It was noticed that Bubble's mission cannot be performed by a mobile device for 60,000 or more arrays and thus no output is seen in the graph. The findings of cloud execution for Wi-Fi and 3G are somewhat close. In order to obtain an understanding of how cloud computing operates on the mobile device, the results extract connectivity and measurement time. The bulk of the contact time has been spent on the transfer of data and not on interaction. Therefore the contact time with the growing array has no substantial effect on the total migratory output benefit and remains many times greater than the performance of mobile devices.

Conclusion:

A decision-making algorithm was suggested and applied in this paper to decide whether a task can be transferred to a cloud server for computer-intensive activities. The findings suggest that cloud performs mobile devices independent of network connectivity technology for tasks of strongly measured and limited data transfer. In the future, the authors consider various types of tasks by expanding the algorithm to operate with many virtual machines and running task modules in parallel to further minimize the time required for completing multiple tasks.

Acknowledgement:

The authors express gratitude towards the assistance provided by The Management, Krupanidhi Group of Institutions (KGI) and Krupanidhi Research Incubation Centre, KGI in completing the research. We also thank our Research Mentors who guided us throughout the research and helped us in achieving the desired results.

References:

1. Mark Billinghurst and Thad Starner. Wearable devices: new ways to manage information. Computer, 32(1):57–64, 1999.
2. Global mobile app usage statistics. https://www.statista.com/topics/1002/mobile-app-usage/, Access Date: 10 May 2017.
3. Hoang T Dinh, Chonho Lee, Dusit Niyato, and Ping Wang. A survey of mobile cloud computing: architecture, applications, and approaches. Wireless communications and mobile computing, 13(18):1587–1611, 2013.
4. Junjie Peng, Yongchuan Dai, Yi Rao, Xiaofei Zhi, and Meikang Qiu. Modeling for cpu-intensive applications in cloud computing. In High Performance Computing and Communications (HPCC), 2015 IEEE 7th International Symposium on Cyberspace Safety and Security (CSS), 2015 IEEE 12th International Conference on Embedded Software and Systems (ICESS), 2015 IEEE 17th International Conference on, pages 20–25. IEEE, 2015.
5. EC Amazon. Amazon elastic compute cloud (amazon ec2). Amazon Elastic Compute Cloud (Amazon EC2), 2010.
6. B. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "Clonecloud: Elastic Execution between Mobile Device and Cloud," Proc. sixth Conf. Comput. Syst., pp. 301–314, 2011.
7. Sokol Kosta, Andrius Aucinas, Pan Hui, Richard Mortier, and Xinwen Zhang. Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code off loading. In Infocom, 2012 Proceedings IEEE, pages 945–953. IEEE, 2012.
8. Marco V Barbera, Sokol Kosta, Alessandro Mei, and Julinda Stefa. To offload or not to offload? The bandwidth and energy costs of mobile cloud computing. In INFOCOM, 2013 Proceedings IEEE, pages 1285–1293. IEEE, 2013.
9. Karthik Kumar, Jibang Liu, Yung-Hsiang Lu, and Bharat Bhargava. A survey of computation offloading for mobile systems. Mobile Networks and Applications, 18(1):129– 140, 2013.
10. Mazliza Othman, Sajjad Ahmad Madani, Samee Ullah Khan, et al. A survey of mobile cloud computing application models. IEEE Communications Surveys & Tutorials, 16(1):393–413, 2014.
11. Zixue Cheng, Peng Li, Junbo Wang, and Song Guo. Justintime code offloading for wearable computing. IEEE Transactions on Emerging Topics in Computing, 3(1):74– 83, 2015.
12. Puneet Kumar and Er Manwinder Singh. 5g technology of mobile communication. International Journal of Electronics and Computer Science Engineering, 2(4):1265– 1275, 2012.
13. Eldad Perahia, Carlos Cordeiro, Minyoung Park, and L Lily Yang. Ieee 802.11 ad: Defining the next generation multi-gbps wi-fi. In Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE, pages 1–5. IEEE, 2010.
14. Oscar Bejarano, Edward W Knightly, and Minyoung Park. Ieee 802.11 ac: from channelization to multi-user mimo. IEEE Communications Magazine, 51(10):84–90, 2013.
15. Mahbub E Khoda, Md Abdur Razzaque, Ahmad Almogren, Mohammad Mehedi Hassan, Atif Alamri, and Abdulhameed Alelaiwi. Efficient computation offloading decision in mobile cloud computing over 5G Network. Mobile Networks and Applications, 21(5):777–792, 2016.
16. Viji Vinod, Rajkumar, N, Karthikeyan, S & Subramanian, C 2015, 'Expectation of Rising Customer Intelligence System in Road Service Transport using Cloud Services', International Journal of Applied Environmental Sciences, vol.10, no.1, pp. 135-142.
17. Nirmala Sugirtha Rajini, S, Rajkumar, N & Mercy Beulah, E 2015, 'An Innovative Cloud Computing Infrastructure for e-Tourism', International Journal of Applied Environmental Sciences, vol.10, no.1, pp. 19-22.
18. Gohin, B, Viji Vinod & Rajkumar, N 2012, 'Cloud Computing Architecture for Visual Disabilities on E-Governance', International Journal of Computer Applications, vol. 52, no. 1, pp. 44.48.
19. Abinaya, S, Rajkumar, N, Josephine, MS & Karthikeyan, S 2016, 'A study on Keyword ranked Searching and Encrypted over mobile Cloud Data', Proceedings of the National conference on Smart Information Light fidelity (Li-fi) Internet of things Conferences (Silicon'16), Dr. MGR Educational and Research Institute University, Chennai
20. Gohin, B, Viji Vinod & Rajkumar, N 2013, 'Evaluation of Cloud software service accessibility of E-Governance Websites', International Conference (ICTCST-2013), Bakthavatsalam Memorial College for Women, Chennai, pp. 112-116.
21. Gohin, B, Viji Vinod & Rajkumar, N 2012, 'Usability of Cloud SaaS based E-Governance for people with Disabilities', International Journal of Computer Applications Proceedings on E-Governance and Cloud Computing Services – 2012, vol. 2, no. 2, pp. 23-28.
22. Mercy Beulah E, Nirmala Sugirtha Rajini S & Rajkumar N 2016, 'Application of Data Mining in Healthcare: A Survey', Asian Jr. of Microbiol. Biotech. Env. Sc., Vol. 18, No. 4, pp. 999-1001.
23. Athitya Kumaraguru, M, Viji Vinod & Rajkumar, N 2017, 'A Comparative Study on SVM and RVM in Imbalance Data', Journal of Advanced Research in Dynamical & Control Systems, vol. 10, no.03-Special Issue pp. 1393-1395.
24. Mercy Beulah E, Nirmala Sugirtha Rajini S & Rajkumar N 2016, 'Application of Data Mining in Healthcare: A Survey', Asian Jr. of Microbiol. Biotech. Env. Sc., Vol. 18, No. 4, pp. 999-1001.