RESEARCH ARTICLE OPEN ACCESS

Techniques to Minimize State Transfer Cost for Dynamic Execution Offloading In Mobile Cloud Computing

S.Rathnapriya[1], S. Sumithra[2]
[1](Department of Electronics and Communication, Pavendhar Bharathidasan College of Engineering and Technology, Anna University, Trichy-24)
[2] (Department of Electronics and Communication, Pavendhar Bharathidasan College of Engineering and Technology, Anna University, Trichy-24)

ABSTRACT
The recent advancement in cloud computing in cloud computing is leading to and excessive growth of the mobile devices that can become powerful means for the information access and mobile applications. This introducing a latent technology called Mobile cloud computing. Smart phone device supports wide range of mobile applications which require high computational power, memory, storage and energy but these resources are limited in number so act as constraints in smart phone devices. With the integration of cloud computing and mobile applications it is possible to overcome these constraints by offloading the complex modules on cloud. These restrictions may be alleviated by computation offloading: sending heavy computations to resourceful servers and receiving the results from these servers. Many issues related to offloading have been investigated in the past decade.
Keywords-Cloud computing, Complex modules, Execution offloading ,Mobile computing, Smart Phones.

I. INTRODUCTION

Cloud computing delivers infrastructures, platform and software that provides as services on the usage based payment model to end users. These services are Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Mobile Cloud Computing at its simplest refers to an infrastructure where both the data storage and the data processing happen outside of the mobile device. Mobile Cloud applications move the computing power and data storage away from mobile phones and into the cloud, bringing applications and mobile computing to not just Smartphone users but a much broader range of mobile subscribers. It is a productive business choice that transfers data from smartphone devices to powerful and centralized computing platforms located in the cloud. Thus reducing the development and running cost of mobile applications on smartphone devices. This technique consumes fewer resources of Smartphone devices making them more efficient. The architecture of MCC is such that the various mobile devices are connected to their respective mobile networks via base station (BTS, Access Points or Satellite). The requests made by users are transferred to servers providing mobile network services. Now the mobile network operator can provide services to mobile users as authentication, authorization and accounting based on data stored in database. After authorizing user the requests are transferred to cloud via internet.

Advancements in computing technology have expanded the usage of computers from desktops and mainframes to a wide range of mobile and embedded applications, including surveillance, environmental sensing, GPS navigation, mobile phones, autonomous robots, etc. Many of these applications run on systems with limited resources. For example, mobile phones are battery powered. Environmental sensors have small physical sizes, slow processors, and small amounts of storage. Most of these applications use wireless networks and their bandwidths are orders-of-magnitude lower than wired networks. Meanwhile, increasingly complex programs are running on these systems—for example, video processing on mobile phones and object recognition on mobile robots. Thus there is an increasing gap between the demand for complex programs and the availability of limited resources.

*Off loading*is a solution to augment these mobile systems' capabilities by migrating computation to more resourceful computers (i.e., servers). This is different from the traditional client-server architecture, where a thin client *always* migrates computation to a server. Computation offloading is also different from the migration model used in multiprocessor systems and grid computing, where aprocess may be migrated for load balancing. The key difference is that computation offloading migrates programs to servers outside of the users' immediate computing environment; process migration for grid computing typically occurs from one computer to another within the same computing environment, i.e., the grid. A significant amount of research has been performed on computation offloading: making it feasible, making offloading decisions, and developing offloading infrastructures. This was primarily due to limitations in wireless networks, such as low bandwidths. Improvements in virtualization technology, network bandwidths, and cloud computing infrastructures, have shifted the direction of offloading. These developments have made computation offloading more practical.

Offloading may save energy and improve performance on mobile systems. However, this usually depends on many parameters such as the network bandwidths and the amounts of data exchanged through the networks. Many algorithms have been proposed to make offloading decisions to improve performance or save energy. Offloading requires access to resourceful computers for short durations through networks, wired or wireless. Cloud computing allows *elastic* resources and offloading to multiple servers; it is an enabler for computation offloading. Various infrastructures and solutions have been proposed to improve offloading: they deal withvarious issues such as transparency to users, privacy, security, mobility, etc.

II. PROPOSED SYSTEM

Our enhanced proposed system enables multi client communication with cloud server, which efficiently offloading the job between client and server. Here low overhead, energy consumption and execution cost is takes as a key attribute. The overall process is build based on to achieve these attributes in a minimized transfer cost. To achieve this idea mobile client, intermediate node, and cloud server is implemented. A portion of a client request is initially transferred to the intermediate node, this node will transcode the request to the cloud. During the transcoding process the intermediate node serves asa clone cloud to the mobile client and reduces the overhead at both sides of client and the server.

III. MODULES

- Route Selection Module
- Send Request Module
- Route Forwarding Module
- Distance Calculation Module
- Display Route and PerformanceEvaluation Module
- Multi Communication module

3.1 ROUTE SELECTION MODULE

Route selection module enables the user to choose the source and destination. Choosen source and destination is sending to the user side application. This is used for choosing the possible way of a source and destination.

3.2 SEND REQUEST MODULE

Send Request modulesends the source and destination value to the server side application to find the possible route option. This is helps to partition the work with the server. By partitioning the work itreduces the execution time from the overall computation time

3.3 ROUTE FORWARDING MODULE

Route Forwarding module is a server side module which provides and satisfies the user expectation based on their request. When the user request for possible routes it first receives the source and destination values and check for the possible routes. Then the computed possible route values are sending back to the user. From the collection of possible route values user selects their desired route value and ask for the distance of that route to the server.

3.4 DISTANCE CALCULATION MODULE

At Distance Calculation Module, the server receives the possible route selected by the user and responds the distance value back to them. Then, the application in the user side computes the overall time taken to travel from source to destination.

3.5 DISPLAY ROUTE&PERFORMANCE MODULE

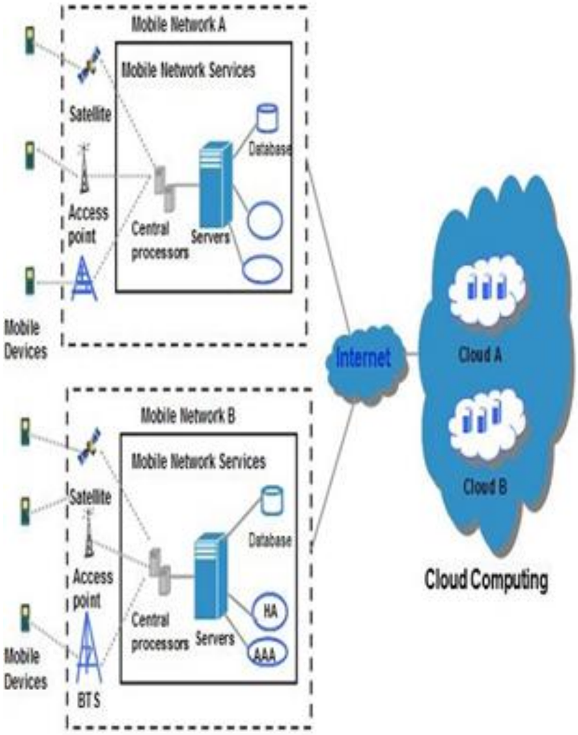
This module generates a route map for the selected possible route value. Calculate time consumption and consumption. Finally, calculate performance evaluation.

3.6 MULTI CLIENT COMMUNICATION MODULE

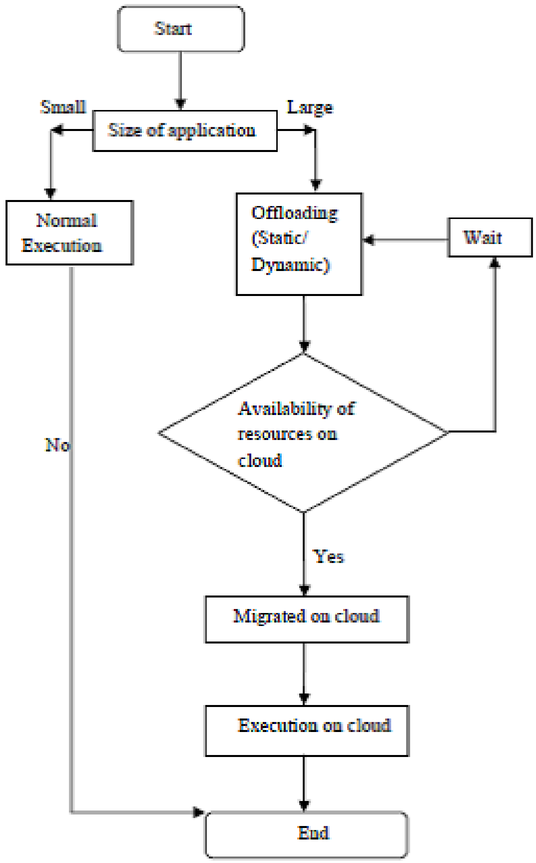
Multi client communication module enhances the existing process by implementing multi client communication to the cloud server for servicing the request. This module utilizes client process, intermediate node and server process. The client process request for a service to the cloud server then the intermediate node will serve portion of the request at its side and forward the request to the server then it will be processed by the server and sent back to the client. During this process the overall transmission time and execution cost is calculated.

IV. FIGURES

4.1 ARCHITECTURE DIAGRAM



4.2 FLOW DIAGRAM



V. CONCLUSION

Mobile Cloud Computing is the integration of cloud computing with smartphone devices. It provides rich communication between cloud-mobile users and cloud providers regardless of heterogeneous environments. Offloading is the method to migrate the complex modules of the application to the cloud. All the complicated computations are performed on the cloud and the results are sent back to the smartphone device. It analysis the energy consumption in a smartphone when executing a computation intensive task and end-to-end energy consumption when the same task is offloaded to a remote server.

ACKNOWLEDGEMENT

I would like to thank my shepherd and the anonymous reviewers for their suggestion and feed back.

REFERENCES

[1] E. Cuervo, A. Balasubramanian, D.-k. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "*MAUI: Making Smartphones Last Longer with Code Offload,"* Proc. ACM Eighth Int'l Conf. Mobile Sys-tems, Applications, and Services (MobiSys) , 2010.

[2] M.-R. Ra, A. Sheth, L. Mummert, P. Pillai, D. Wetherall, and R.Govindan, "*Odessa: Enabling Interactive Perception Applicationson Mobile Devices,*" Proc. ACM Ninth Int'l Conf. Mobile Systems,Applications, and Services (MobiSys), 2011.

[3] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "*CloneCloud: Elastic Execution between Mobile Device and Cloud*," Proc. ACM Sixth Conf. Computer Systems (EuroSys) , 2011.

[4] I. Giurgiu, O. Riva, D. Juric, I. Krivulev, and G. Alonso, "*Calling the Cloud: Enabling Mobile Phones as Interfaces to Cloud Applications*," Proc. ACM/IFIP/USENIX 10th Int'l Conf. Middleware (Middleware) , 2009.

[5] R. Kemp, N. Palmer, T. Kielmann, and H.E. Bal, "*Cuckoo: A Computation Offloading Framework for Smartphones*," Proc. Second Int'l ICST Conf. Mobile Computing, Applications, and Services (MobiCASE), 2010.

[6] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "*The Case for VM-Based Cloudlets in Mobile Computing*," IEEE Pervasive Computing, vol. 8, no. 4, pp. 14-23, Oct.-Dec. 2009.

[7] D. Gavalas and D. Economou, "*Development Platforms for Mobile Applications: Status and Trends*," IEEE Software, vol. 28, no. 1, pp. 77-86, Jan./Feb. 2011

[8] R. Ma and C.-L. Wang, "*Lightweight Application-Level Task Migration for Mobile Cloud Computing,*" Proc. IEEE 26th Int'l Conf. Advanced Information Networking and Applications (AINA), 2012.

[9] S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, "*ThinkAir: Dynamic Resource Allocation and Parallel Execution in the Cloud for Mobile Code Offloading*," Proc. IEEE INFOCOM, 2012

[10] D. Kovachev, T. Yu, and R. Klamma, "*Adaptive Computation Off-loading from Mobile Devices into the Cloud*," Proc. IEEE 10th Int'l Symp. Parallel and Distributed Processing with Applications (ISPA), 2012.

[11] R. Newton, S. Toledo, L. Girod, H. Balakrishnan, and S. Madden, "*Wishbone: Profile-Based Partitioning for Sensornet Applications*," Proc. Sixth USENIX Symp. Networked Systems Design and Implementation (NSDI), 2009.