A Study on Cloud Based Augmentation in Mobile Devices

Anjima Ajith
PG scholar/Department of IT,
Karunya University, Tamilnadu, India.

S. Durga
Assistant Professor/ Department of IT,
Karunya University, Tamilnadu, India.

Abstract

Nowadays the mobile devices are widely used and recent advancement in the field of wireless network put forwards to the access of the mobile internet. The resource-constrained nature of the mobile devices led to the limitations for the improvements in mobile applications and services. Thus the cloud based augmentation approaches has been remarkably increased to expand their computational capabilities towards the cloud. Many issues related to cloud based augmentation are investigated in the past decade. Here in this survey we come up with the methodology, background and some research areas of the cloud based augmentation.

Key Words: **Mobile cloud computing, computation offloading, augmentation.**

1. Introduction

In the past few decades we have come across the advancement in the computing technology from the desktop to the wide range of mobile and embedded applications. Since the mobile devices are battery powered, they have limited resource compared to the computers. In order to enhance the computation power of the mobile devices, the resource rich applications are migrated to the cloud. Thus the cloud based augmentation approached came into existence.

Cloud computing as "a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service or service provider interaction" [1] stimulates researchers to adopt a new technology in mobile device augmentation: Cloud-based Mobile Augmentation (CMA). Cloud-based Mobile Augmentation (CMA) is the-state-of-the-art mobile augmentation model that leverages cloud computing technologies and principles to increase, enhance, and optimize computing capabilities of mobile devices by executing resource intensive mobile application components in the resource-rich cloud-based resources.

The purpose of this paper is to make the readers understands the various Research areas on cloud based augmentation for mobile systems. This paper provides an overview of the techniques, methodology, and architectures for cloud augmentation. The survey is based on some common approaches used to make offloading decisions including,

- Why to offload (improve performance or save energy)
- When to make the decision for offloading (static vs dynamic)
- Infrastructures for offloading (grid and cloud computing).

In this paper, *many* studies have been conducted on topics related to cloud based augmentation and a comprehensive survey of all studies would be impossible. Hence, this paper does not illustrate a complete survey on this area. The references are selected based on our limited knowledge of the topics.

2.1 A study on virtual machine deployment for application outsourcing in mobile cloud computing

In this paper, Muhammed Shiraz et.al [2] tells about offloading the computational load to cloud server nodes. In order to offload the computational resources here they ensure that VM deployment and management requires additional computing resources on mobile device. This paper analyzes the impact of VM deployment and management on the execution time of application. They have investigated on VM deployment and management for application processing in simulation environment by using CloudSim. CloudSim which is a simulation toolkit that provides a simulation framework to model the working of VM deployment and management for

application processing in cloud-computing infrastructure.
The major advantages is this work is that it can save power and have an efficient bandwidth utilization and they support User preferences and it provides less execution cost. The disadvantages are the additional overhead for deployment and they utilize more computing resources and execution time is increased.

2.2 CloneCloud

CloneCloud is a fine grained, thread-level, which is cloud-based, application partitioner and execution runtime that migrates entire mobile platform into the cloud VM. It runs the mobile application inside the VM without performing any change in the application code. The clonecloud allow the local execution of remaining mobile application when remote server is running the intensive components. Cloud resources simulate distributed execution of an application in a resourceful environment without engaging application developer into the distributed application programming domain.
The disadvantage of this paper is the Communication overhead, the frequent synchronization of the shared data between the mobile and cloud. Shrinking the power of cloud is another disadvantage in this paper. The advantages are data encapsulation and migration, mobile-cloud data synchronization .Increase the communication traffic is another advantages and it has increased execution time and the energy efficiency of the offloading.

2.3 The case of VM-based cloudlets in mobile computing

Virtual machine (VM) technology is been exploited in this architecture to rapidly instantiate customized service software on a nearby *cloudlet* and by using these services over a wireless LAN.
The mobile device will functions like a thin client with respect to the service. A cloudlet is a trusted, resource-rich computer or cluster of computers that's available for use by nearby mobile devices by connecting to the internet.
Using a cloudlet also simplifies the challenge of meeting the peak bandwidth demand of multiple users interactively generating and receiving media such as high-definition video and high-resolution images. Rapid customization of infrastructure for diverse applications emerges as a critical requirement, and our results from a proof-of-concept prototype suggest that VM technology can indeed help meet this requirement. Here we have to focus in reducing the WAN delays, Congestion and Failures occurring in the cloudlets.

2.4 Tactic based remote execution for mobile computing

In this paper, the author introduce a compact declarative form called Tactic in which automated dynamic repartitioning of mobile applications can be reconciled and the useful knowledge about an application relevant to remote execution is been captured. Chroma is a tactic based remote execution system that performs comparably to a runtime system that makes perfect partitioning decisions. Here they also show that Chroma can automatically utilize extra resources in an overprovisioned environment to improve application performance.
In this paper, they have studied on three applications of the above technique (natural language translation, speech recognition, and face recognition) and show that the tactics for each is much less than one percent of total code size. In addition, Chroma can optionally utilize extra resources in an over-provisioned environment and allows us to achieve lower latencies for the three applications.

2.5 Cuckoo: a Computation Offloading Framework for Smartphone

The aim of this paper is to enhance the performance of smartphone application by reducing the energy usage. The author propose a Cuckoo framework by the development of smartphone applications that can undergo computation offloading and provides a dynamic runtime system, which decide at runtime, whether a part of an application will be executed locally or remotely. It is targeted at the Android platform. A resource manager application for smartphone users and a programming model for developers, which integrates into the Eclipse build system. It provides maximum computation speed and minimum energy usage.
There are some limitations in this paper. Here Cuckoo does not yet support callbacks. It does not support any form of security. It supports only stateless service. Cuckoo can do both early and late binding to remote resources.

2.6 NWSLite: A light-weight prediction utility for mobile devices

In this paper, the authors propose a new concept called NWSLite which is a computationally efficient, highly accurate prediction utility for mobile devices. Network

Weather Service (NWS) can be defined as a dynamic forecasting toolkit for adaptive scheduling of high-performance Computative applications. They have significantly reduce the NWS to reduce its resource consumption yet still achieve accuracy that exceeds that of extant remote execution prediction methods. There are some advantages in using this NWS Lite is that high prediction accuracy, Consumes less computation resources and provides lower cost.

2.7 Using History to Improve Mobile Application Adaptation

In this paper, the researchers use historical application logging data to predicate the fidelity of an application. Fidelity is the application-specific notion of the goodness of a computed result or data object and also decides its resource consumption. A lower fidelity result in lower resource consumption.
The ultimate goal of fidelity adaptation is to improve a mobile user's computing experience by delivering result quickly, with low battery drain and little destruction of the user. Augmented Odyssey, an operating system platform for adaptation, with a history based prediction system that monitors, logs and predicts application resource consumption as a function of the fidelity.
Our current prototype has a CPU overhead of 0.22% for a typical application. The disadvantage is that acquired history logs for only hardware platform that they might ever use.

2.8 Adaptive Offloading Inference for Delivering Applications in Pervasive Computing Environments

By using the Fuzzy Control model, they have developed an offloading inference engine to adaptively solve two key decision-making problems during runtime offloading such as timely triggering of adaptive offloading, and selection of an application partitioning policy.
Extensive trace-driven evaluations show the effectiveness of the offloading inference engine. In this paper, we present the offloading inference engine (OLIE), which makes decisions to enable AIDE to deliver applications on resource-constrained mobile devices with minimum overhead.
There are some disadvantages in this paper.
Here it is based only upon the memory with lower overhead than other approaches. It is not considering the following multiple factors such as costs, device status, and cloud status/usage and user preference.

2.9 The case of cyber foraging

In this paper R. Balan et.al [10] explore the concept of cyber foraging and discuss specifically on how the use of surrogates can help in two distinct situations. First, we demonstrate how it can reduce cache miss service times in mobile file access. Second, we show how it enables compute-intensive applications like language translation and augmented reality to run on mobile hardware.
Cyber foraging is a common approach explored by many to augment the capability of resource-constrained mobile devices. The basic idea is to dynamically discover and make use of nearby resources, aka surrogates, to offload the execution of an application or parts of an application running on a mobile device. The advantage of this research is increased resource availability and dynamic remote execution. The disadvantage is inconsistent and they require application developer to modify the application.

2.10 On Effective offloading services for resource-constrained mobile devices running heavier mobile internet application

In this paper, the author proposes an offloading technique which reduces the consumption of CPU, memory and communication cost. It supports the mobility of the resource intensive application and makes use of multiple surrogates. The main objective is that it reduces the response time for a user application. To minimize the overall traffic overhead between the mobile device and the surrogates, they transform the application into classes and will be decided whether to offload according to their class weights. Application partitioning is done to compare the class weights of the classes that are to be connected to the one hop neighboring classes. Security and privacy in terms of both the surrogates and offloaded application is not been considered in this paper.

2.11 MAUI: Making Smartphones last longer with code offload

MAUI is fine-grained energy aware offloading technique in which the application is executed either locally in the mobile devices or remotely migrated to the cloud. It using code portability method to maximize the energy saving. The authors focus on the programming reflection and make use of serialization to determine its networking costs. The application is partitioned dynamically in this paper. There are some limitations in this method. While using the power-save

mode, it can hurt the overall energy consumption. It takes a considerable amount of time for programmer to annotate offloadable methods.

2.12 A Virtual Cloud Computing Provider for Mobile Devices

By reducing the heterogeneity impact of the hardware and platform, in this paper, G. Huerta-Canepa et.al [13] aims to extend the computation capabilities of mobile devices to an ad-hoc cluster of nearby smartphones.
The architecture focuses on offloading the application from cloud computing provider client towards an ad-hoc mobile cloud framework using the RPC method implemented by Jabber RPC. When the application is decided to be offloaded, security and privacy requirement and offloading overhead is considered. The disadvantage of this paper is mobility of application is not considered. Increase in offloading time and overhead since computing job is sent to one mobile node.

2.13 Towards an elastic application model to augment the computation capabilities of mobile devices with cloud computing

The author here proposed an elastic framework that uses the cloud resources to execute the resource-intensive component of the mobile device. In this model, they created Weblets by partitioning the mobile application into small components. In order to increase system robustness by decreasing the communication overhead and latency, the author made the weblets with least dependency with each other. The weblet execution is dynamically configured to either perform locally or remotely, based on the weblet's resource partitioning, execution configuration quality, and criteria of offloading. The advantage in this paper is that in order to enhance the overall execution performance and enrich user experience, the system is able to execute the weblets both locally and remotely. Elastic application model pays more attention to the user preferences by enabling different running modes of a single application.

Comparison table

Methods	CPU Usage	Memory	Throughput	Cost	Battery Prolonging
VM migration[2]	medium	medium	high	low	low
CloneCloud[3]	medium	medium	medium	low	low
Cloudlet[4]	medium	medium	high	low	medium
Tactic Based[5]	low	low	medium	medium	Not applicable
Cuckoo[6]	low	medium	high	low	medium
NWSLite[7]	medium	medium	medium	low	medium
Adaptive Offloading[8]	medium	medium	medium	low	low
Using history[9]	low	low	medium	low	medium
Cyber foraging[10]	low	medium	medium	Not applicable	Not applicable

Effective offloading services[11]	low	low	medium	low	medium
MAUI[12]	medium	medium	high	medium	low
VMCC[13]	medium	medium	high	medium	medium
Elastic application[14]	medium	medium	high	low	high

Conclusion

This paper survey on wide research area of the cloud based offloading for mobile devices like smart phones, laptop, tablets etc. here we survey on different variety of algorithm to migrate the resource intensive computations towards the cloud. Finally in accordance with our knowledge the paper on elastic application framework shows better performance than rest of the papers in this survey.

References

[1] P. Mell and T. Grance, "The NIST Definition of Cloud Computing Recommendations of the National Institute of Standards and Technology," 2011.
[2]Muhammed Shiraz, Saeid Ablofazli, Zohreh Sanaei, Abdullah Gani "A study on virtual machine deployment for application outsourcing in mobile cloud computing. In springer,2012.
[3] B. G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "CloneCloud: Elastic execution between mobile device and cloud," in *Proc. ACM EuroSys'11*, Salzburg, Austria, 2011, pp. 301–314.
[4]M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies. The case for vm-based cloudlets in mobile computing. *IEEE Pervasive Computing*, (4), 2009.
[5]R. K. Balan, M. Satyanarayanan, S. Park, and T. Okoshi, "Tactics-based remote execution for mobile computing." in Proc. ACM MobiSys'03, San Francisco, California, USA, 2003, pp. 273–286.
[6]Roelof Kemp, Nicholas Palmer, Thilo Kielmann and Hentri Bal "Cuckoo: A Computation offloading Framework for Smartphones"in Springer,2012.
[7]Gurun S, Krintz C, Wolski R (2004) "NWSLite: a lightweight prediction utility for mobile devices." In: International conference on mobile systems, applications, and services, pp 2–11
[8] X. Gu, K. Nahrstedt, A. Messer, I. Greenberg, and D. Milojicic. "Adaptive offloading inference for delivering applications in pervasive computing environments." In Proc. of IEEE International Conference on Pervasive Computing and Communications, 2003.
[9] D. Narayanan, J. Flinn, and M. Satyanarayanan. "Using history to improve mobile application adaptation." In Proc. of the 3rd IEEE Workshop on Mobile Computing Systems and Applications, 2000.
[10] R. Balan, J. Flinn, M. Satyanarayanan, S. Sinnamohideen, and H. Yang, "The case for cyber foraging." In *Proc. of the 10th ACM SIGOPS European Workshop*, 2002.
[11] Kun Yang and Shumao Ou, University of Essex Hsiao-Hwa Chen ," On Effective Offloading services for resource-constrained mobile devices running heavier mobile internet applications" In Proc. Of IEEE communications Magazine, Jan2008.
[12] Cuervo E, Balasubramanian A, Cho D, Wolman A, Saroiu S, Chandra R, Bahl P (2010) MAUI: making smartphones last longer with code offload. In: International conference on mobile systems, applications, and services, pp 49–62
[13] G. Huerta-Canepa and D. Lee, "A Virtual Cloud Computing Provider for Mobile Devices," in *Proc. ACM MCS'10: Social Networks and Beyond*, San Francisco, USA, 2010, pp. 1–5.
[14] X. W. Zhang, A. Kunjithapatham, S. Jeong, and S. Gibbs, "Towards an Elastic Application Model for Augmenting the Computing Capabilities of Mobile Devices with Cloud Computing," *Mobile Networks & Applications*, vol. 16, no. 3, pp. 270–284, 2011.