Chapter 8
Accelerating Mobile-Cloud Computing:
A Survey

Tolga Soyata
University of Rochester, USA

Wendi Heinzelman
University of Rochester, USA

He Ba
University of Rochester, USA

Minseok Kwon
Rochester Institute of Technology, USA

Jiye Shi
UCB Pharma, UK

ABSTRACT

With the recent advances in cloud computing and the capabilities of mobile devices, the state-of-the-art of mobile computing is at an inflection point, where compute-intensive applications can now run on today's mobile devices with limited computational capabilities. This is achieved by using the communications capabilities of mobile devices to establish high-speed connections to vast computational resources located in the cloud. While the execution scheme based on this mobile-cloud collaboration opens the door to many applications that can tolerate response times on the order of seconds and minutes, it proves to be an inadequate platform for running applications demanding real-time response within a fraction of a second. In this chapter, the authors describe the state-of-the-art in mobile-cloud computing as well as the challenges faced by traditional approaches in terms of their latency and energy efficiency. They also introduce the use of cloudlets as an approach for extending the utility of mobile-cloud computing by providing compute and storage resources accessible at the edge of the network, both for end processing of applications as well as for managing the distribution of applications to other distributed compute resources.

DOI: 10.4018/978-1-4666-4522-6.ch008

INTRODUCTION

Recent developments in mobile computing have truly empowered human users, as mobile computing can augment cognitive capabilities dramatically, e.g., through voice recognition, natural language processing, machine learning, augmented reality, and decision-making (Satyanarayanan et al., 2009). With recent advances in mobile devices, coupled with the technological advances in wireless and cloud technologies, computationally intensive applications can now run on devices with limited resources such as tablets, netbooks and smartphones using the cloud remotely as an additional computational resource.

Although different definitions exist in the literature (Dinh, et al, 2011; Fernando et al., 2013), we define mobile-cloud computing as the co-execution of a mobile application within the expanded mobile/cloud computational platforms to optimize an objective function. A typical objective function is the *application response time*, where the goal is to minimize the objective function. Expanding the application computational resources beyond the mobile is necessary for applications where the objective function cannot be minimized sufficiently by the mobile platform alone (e.g., real-time face recognition), as well as for applications that rely on data not stored on the mobile device. In mobile-cloud computing, it is crucial to provide the user seamless, transparent and cost-effective services as mobile devices rent computing, storage, and network resources from the cloud in order to process and store a vast amount of data (AWS, 2012; Microsoft, 2012; Google, 2012). (AWS, 2012)(Microsoft, 2012)(Google, 2012)

We define *application cost* as an example objective function that quantifies the fees charged by Cloud operators, such as Amazon Web Services, during the execution of the application. For example, Amazon charges for compute-usage per hour per CPU instance, which implies increasing application costs as the required amount of computation increases. Similarly, cloud operators charge for the usage of database instances, such as Microsoft SQL Server. Table 1 shows some example mobile-cloud applications and their computational/storage demands, as well as their application response-time sensitivity. While applications requiring higher computational and storage resources might cost more during operation in a Cloud platform such as AWS, certain response-time sensitive applications, such as the Battlefield application described in Table 1, might tolerate this increased cost due to their need for low response time. Notice that Cloud operators charge less for compute-resources with lower response time guarantees. Specifically, while AWS charges nothing for *Micro* instances with *no* response time guarantees, it charges a small amount for the *Small* instance, and significantly higher for the *Large* instance, which is a dedicated CPU instance. By the preparation of this document, the AWS pricing for these instances ranged from $0.10 to $0.40 per core per GHz per hour (AWS, 2012), where the unit price decreased with a higher core-count commitment (i.e., number of cores available to an instance). This implies a rich variety of options when executing mobile-cloud applications. The choice of the Cloud CPU instances depends on the application priorities listed in Table 1.

The primary focus of this chapter is to elaborate on the techniques that enable these mobile-cloud applications to achieve the goals listed in Table 1. Although the demands of these applications will not change from that shown in this table, achieving certain goals might never become possible by using mobile-only or even a mobile-cloud combination. This is due to the limited computation and storage on a mobile device, which does not permit the processing or storage of large amounts of data locally, as well as the high network latencies connecting the mobile and cloud, plac-

Related Content

An Adaptive Service Monitoring System in a Cloud Computing Environment

Karthikeyan P.and Sathiyamoorthy E. (2020). *International Journal of Grid and High Performance Computing (pp. 47-63).*

www.irma-international.org/article/an-adaptive-service-monitoring-system-in-a-cloud-computing-environment/249743

An Online Service Performance Prediction Learning Method

Hua Liangand Sha Wang (2022). *International Journal of Grid and High Performance Computing (pp. 1-14).*

www.irma-international.org/article/an-online-service-performance-prediction-learning-method/301577

Rough Set-Based Feature Selection: A Review

Richard Jensen (2008). *Rough Computing: Theories, Technologies and Applications (pp. 70-107).*

www.irma-international.org/chapter/rough-set-based-feature-selection/28468

A Comparative Study and Algorithmic Analysis of Workflow Decomposition in Distributed Systems

Ihtisham Aliand Susmit Bagchi (2019). *International Journal of Grid and High Performance Computing (pp. 71-100).*

www.irma-international.org/article/a-comparative-study-and-algorithmic-analysis-of-workflow-decomposition-in-distributed-systems/216482

Pre-Cutoff Value Calculation Method for Accelerating Metric Space Outlier Detection

Honglong Xu, Zhonghao Liang, Kaide Huang, Guoshun Huangand Yan He (2024). *International Journal of Grid and High Performance Computing (pp. 1-17).*

www.irma-international.org/article/pre-cutoff-value-calculation-method-for-accelerating-metric-space-outlier-detection/334125