TASK OFFLOADING BETWEEN SMARTPHONES AND DISTRIBUTED COMPUTATIONAL RESOURCES

by

Shigeru Imai

A Thesis Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the degree of

MASTER OF SCIENCE

Major Subject: COMPUTER SCIENCE

Approved:

_____

Carlos A. Varela, Thesis Adviser

Rensselaer Polytechnic Institute
Troy, New York

May, 2012

ABSTRACT

Smartphones have become very popular. While people enjoy various kinds of applications, some computation-intensive applications cannot be run on the smartphones since their computing power and battery life are still limited. We tackle this problem from two categories of applications. Applications in the first category are single-purpose and moderately slow (more than 10 seconds) to process on a single smartphone, but can be processed reasonably quickly by offloading a single module to a single computer (*e.g.,* a face detection application). Applications in the second category are extremely slow (a few minutes to hours) to process on a single smartphone, but their execution time can be dramatically reduced by offloading computationally heavy modules to multiple computers (*e.g.,* a face recognition application). In this category of applications, management of the server-side computation becomes more important since the intensity of computation is much stronger than in the first category. For the first category, we propose a light-weight task offloading method using runtime profiling, which predicts the total processing time based on a simple linear model and offloads a single task to a single server depending on the current performance of the network and server. Since this model's simplicity greatly reduces the profiling cost at run-time, it enables users to start using an application without pre-computing a performance profile. Using our method, the performance of face detection for an image of 1.2 Mbytes improved from 19 seconds to 4 seconds. For the second category, we present a middleware framework called the *Cloud Operating System* (*COS*) as a back-end technology for smartphones. COS implements the notion of *virtual machine (VM) malleability* to enable cloud computing applications to effectively scale up and down. Through VM malleability, virtual machines can change their granularity by using *split* and *merge* operations. We accomplish VM malleability efficiently by using application-level migration as a reconfiguration strategy. Our experiments with a tightly-coupled computation show that a completely application-agnostic automated load balancer performs almost the same as human-driven VM-level migration; however, human-driven application-level migration outperforms (by 14% in our experiments) human-driven VM-level migration. These results are promising for future fully automated cloud computing resource management systems that efficiently enable truly elastic and scalable workloads.