



Universidade Federal do Ceará
Departamento de Computação
Mestrado e Doutorado em Ciência da Computação

Paulo Antonio Leal Rego

**Applying Smart Decisions, Adaptive Monitoring and
Mobility Support for Enhancing Offloading Systems**

Fortaleza, Ceará
2016

Paulo Antonio Leal Rego

**Applying Smart Decisions, Adaptive Monitoring and
Mobility Support for Enhancing Offloading Systems**

Tese submetida à Coordenação do Programa de Pós-graduação em Ciência da Computação da Universidade Federal do Ceará como parte dos requisitos para obtenção do grau de Doutor em Ciência da Computação.

Orientador: Prof. Dr. José Neuman de Souza

Co-Orientadores: Prof. Dr. Fernando Antonio Mota Trinta e Dr. Masum Z. Hasan

**Fortaleza, Ceará
2016**

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- R268a Rego, Paulo Antonio Leal.
Applying smart decisions, adaptive monitoring and mobility support for enhancing offloading systems /
Paulo Antonio Leal Rego. – 2016.
115 f. : il. color.
- Tese (doutorado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em
Ciência da Computação, Fortaleza, 2016.
Orientação: Prof. Dr. José Neuman de Souza.
Coorientação: Prof. Dr. Fernando Antonio Mota Trinta.
1. Mobile cloud computing. 2. Tomada de decisão de offloading. 3. Mobilidade. 4. Monitoramento
adaptativo. I. Título.

CDD 005

PAULO ANTONIO LEAL REGO

**Applying Smart Decisions, Adaptive Monitoring and Mobility Support for
Enhancing Offloading Systems.**

Tese submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação, da Universidade Federal do Ceará, como requisito para a obtenção do grau de Doutor em Ciência da Computação.

Aprovado em 02 de dezembro de 2016.

BANCA EXAMINADORA

Prof. Dr. José Neuman de Souza
(Presidente)
Universidade Federal do Ceará - UFC

Dr. Masum Z Hasan
(Coorientador)
Cisco Systems - CISCO

Prof. Dr. Fernando Antonio Mota Trinta
(Coorientador)
Universidade Federal do Ceará - UFC

Prof. Dr. Javam de Castro Machado
Universidade Federal do Ceará - UFC

Prof. Dr. Paulo Romero Martins Maciel
Universidade Federal de Pernambuco – UFPE

This work is dedicated to my wife Marcia, my parents Joaquim and Maria Celia, and my siblings Pedro and Vanessa. All I have and will accomplish are only possible due to your love and endless support.

Acknowledgements

This thesis becomes reality with the kind support and help of many individuals. I would like to extend my sincere thanks to all of them.

Foremost, my parents, **Joaquim e Maria Celia**, for the encouragement I had to study and for the love and support at all times of my life. My siblings, **Pedro e Vanessa**, for the unconditional support and partnership. My wife **Marcia**, who is ALWAYS by my side when I need, helped me a lot throughout my years of study. This accomplishment would not have been possible without them.

My advisor, **Prof. Dr. José N. de Souza** for the continuous support throughout my academic life, from undergrad to my Ph.D. study and research. His guidance and friendship helped me in all the time of research and writing of this thesis. My co-advisors, **Prof. Dr. Fernando Trinta** and **Dr. Masum Z. Hasan** for their guidance during the development of this study and for all conversations we had. Thank you all for believing in my potential and for all ideas you gave me.

I would like to thank the **Federal University of Ceará (UFC)** for showing me how to be a student, a professor, and a researcher. The university has given me the opportunity to grow as a person, student and professional. Thanks to all the people who are part of this university, especially my colleagues from the **Quixadá campus**, for the amazing work environment and friendship. I also would like to thank my professors and colleagues from **DC/MDCC**, **GREat**, and other UFC research groups. Thanks for the collaboration and stimulating research environment. It is always a pleasure working with you.

My internship at Cisco System in the US was a unique experience. Thanks, **CNPq**, **Cisco Systems**, and **UFC** for the opportunity. Thanks to all members of the **Cisco CVG team**, your friendship and help were incredible. You all are highly professional and capable.

Last, but not least, I would like to thank **all my friends** who somehow encouraged me to pursue my goals.

“Persistence is the shortest path to success.”
(Charles Chaplin)

Resumo

O hardware de dispositivos móveis tem evoluído nos últimos anos, ao ponto de alguns aparelhos conseguirem alcançar o mesmo desempenho de instâncias de máquinas virtuais. No entanto, apesar dos avanços tecnológicos na capacidade dos smartphones e redes sem fio, a maioria dos dispositivos ainda são computacionalmente limitados se comparados com um computador desktop ou um notebook, e eles enfrentam muitos desafios, principalmente para executar aplicações que requerem computação intensiva. O paradigma *mobile cloud computing* (MCC) estuda formas de estender os recursos computacionais e energéticos dos dispositivos móveis, através da utilização de técnicas de *offloading*. Nesse contexto, esta tese investiga alguns dos desafios identificados na área de MCC, tais como: a decisão de quando e onde fazer *offloading*, a decisão de quais métricas devem ser monitoradas pelo sistema de *offloading*, e o suporte à mobilidade dos usuários em ambientes híbridos, compostos por *cloudlets* e instâncias de nuvens públicas. Diante de tais desafios, esta tese foca no desenvolvimento de um *framework* que permita que aplicações móveis façam *offloading* dinâmico de métodos em um ambiente com múltiplos *cloudlets* e nuvem pública. O *framework* desenvolvido utiliza técnicas de aprendizagem de máquina e redes definidas por software para melhorar a decisão de *offloading*, realizar monitoramento adaptativo e suportar a mobilidade dos usuários. Diversos experimentos foram realizados para avaliar a solução proposta e os resultados mostram que a abordagem desenvolvida para a tomada de decisão é leve e que o serviço de monitoramento adaptativo proposto pode ser utilizado para reduzir o consumo de energia de dispositivos móveis. Além disso, os resultados mostram que a solução proposta pode lidar com diferentes cenários de mobilidade e pode realizar *offloading* em diferentes servidores remotos de forma transparente para o usuário.

Palavras-chaves: *mobile cloud computing*, tomada de decisão de *offloading*, mobilidade, monitoramento adaptativo.

Abstract

The hardware of mobile devices has evolved, and a few device models can even reach the performance of virtual machine instances. Nevertheless, despite technological advances in the capacity of smartphones and wireless technologies, most devices are still computationally limited compared to a desktop computer or a notebook, and they face many challenges to execute applications that require heavy computation. The mobile cloud computing (MCC) paradigm studies how to extend computational resources and the energy of mobile devices through the use of offloading techniques. In this context, this thesis investigates some of the challenges identified in the mobile cloud computing area. Among these challenges, we can mention: the decision of when and where to perform offloading, the decision of which metrics must be monitored by the offloading system, and also the support for user's mobility in a hybrid environment composed of cloudlets and public cloud instances. Given these challenges, this thesis focuses on the development of a framework that allows mobile applications to dynamically perform offloading of methods in a hybrid environment. The developed framework leverages machine learning and software-defined networking techniques to improve offloading decisions, to perform adaptive monitoring, and to support users' mobility. Several experiments were conducted to evaluate the proposed solution, and results show that our offloading decision approach is a lightweight process and the proposed adaptive monitoring service can be used to reduce the energy consumption of mobile devices. Moreover, the results show that the proposed solution supports the most variate mobility scenarios and performs offloading to different remote servers transparently to the user.

Key-words: mobile cloud computing. offloading decision. mobility. adaptive monitoring.

List of Figures

Figure 1 – Public cloud as remote execution environment.	25
Figure 2 – Cloudlet as remote execution environment.	25
Figure 3 – Cluster of mobile devices as remote execution environment.	26
Figure 4 – Hybrid remote execution environment composed of cloudlets, public cloud and a cluster of mobile devices.	26
Figure 5 – Offloading decision trade-off.	28
Figure 6 – Proposed taxonomy for offloading solutions.	33
Figure 7 – SDN view of decoupled control and data planes.	34
Figure 8 – NFV overview.	34
Figure 9 – Overview of the seamless cloud concept.	36
Figure 10 – General approach for building a classification model.	38
Figure 11 – Example of decision tree.	39
Figure 12 – Overview of the MpOS architecture.	52
Figure 13 – Steps to configure MpOS.	54
Figure 14 – Supported scenario: hybrid environment composed of cloudlets and public cloud instances.	57
Figure 15 – Architecture and main components of the proposed solution.	58
Figure 16 – Offloading Dispatcher distributes requests to a proper Application Service.	58
Figure 17 – REE Handler - Exchange of messages between mobile devices and remote servers.	60
Figure 18 – Example of offloading decision tree for a dummy application.	62
Figure 19 – Parameters that comprise the decisions of when and where to offload.	65
Figure 20 – Overview of the components related to the user mobility support.	71
Figure 21 – Example of messages exchanged during a vertical handover. When the mobile device leaves the cloudlet and connects to the public cloud, the cloudlet migrates the Method 2 result to the new remote server.	72
Figure 22 – Tracking users with the Cisco MSE Location API based on XY coordinates.	73
Figure 23 – Experiment 1.1 results - Decision trees' creation time	77
Figure 24 – Experiment 1.2 result - Time needed to deserialize and parse decision trees	79
Figure 25 – Monsoon power monitor	79
Figure 26 – Mobile devices's average power during the BenchImage experiment	82
Figure 27 – Execution time comparison of two observations of Handset A	84
Figure 28 – Mobile devices's average power during the MatrixOperations experiment	85
Figure 29 – Result of the experiment: average FPS of the application	86
Figure 30 – Mobile devices's energy consumption during the CollisionBalls experiment	87

Figure 31 – Decision trees created for the Handset A.	88
Figure 32 – Decision trees created for the Handset B.	89
Figure 33 – Mobile devices’ power consumption during the experiment.	89
Figure 34 – Result of the experiment: mobile devices’s energy consumption	90
Figure 35 – Map of the Cisco campus testbed that was used in the vertical handover experiment. The path starts in the green user symbol (in building SJC- 15) and ends in the red user symbol (in building SJC-13). The yellow circles indicate the coverage area of cloudlets.	91
Figure 36 – Screenshots taken during the Experiment 4.1.	92
Figure 37 – Result of the experiment: execution time of methods with vertical handover	93
Figure 38 – Testbed used in the horizontal handover experiment. The path starts in the green user symbol and ends in the red user symbol. The yellow circles indicate the coverage area of the cloudlets.	93
Figure 39 – Execution time of methods with Cisco MSE disabled and not caching results	94
Figure 40 – Execution time of methods with Cisco MSE enabled and caching results	95
Figure 41 – C2 hierarchical topology	108
Figure 42 – Overview of the MCC Virtual Testbed components.	111
Figure 43 – Dynamic vertical scaling feature. Cpulimit handles the amount of CPU and Libvirt handles the amount of memory allocated to virtual machines.	112
Figure 44 – Overview of devices related to the virtual machine network interface (eth0)	113
Figure 45 – Decision trees created for Handset A.	115
Figure 46 – Decision trees created for Handset B.	115

List of Tables

Table 1 – Confusion matrix for a binary classification problem.	38
Table 2 – Example of training set.	41
Table 3 – Comparison of solutions regarding the offloading aspect	49
Table 4 – Comparison of solutions regarding the decision module	50
Table 5 – Metrics monitored by the Monitoring Service	61
Table 6 – Model Notation	64
Table 7 – Model validation experiments: result of Shapiro-Wilk normality tests	67
Table 8 – Model validation experiments: confidence intervals for all samples	67
Table 9 – Mobile devices’ specification	76
Table 10 – Remote servers’ configuration	76
Table 11 – Experiment 1.1 details	77
Table 12 – Experiment 1.2 details	78
Table 13 – Details of the BenchImage experiment	81
Table 14 – Results of the BenchImage experiment for the Handset A	81
Table 15 – Results of the BenchImage experiment for the Handset B	82
Table 16 – Details of the MatrixOperations experiment	83
Table 17 – Result of the experiment: execution time for different cases on Handset A	83
Table 18 – Result of the experiment: execution time for different cases on Handset B	84
Table 19 – Details of the CollisionBalls experiment	86
Table 20 – Experiment 3 details	88
Table 21 – Configuration of the equipment used in the vertical handover experiment	91
Table 22 – Configuration of the cloudlets used in the horizontal handover experiment	94
Table 23 – Result of the experiment: execution time for different cases	114

List of abbreviations and acronyms

ADSL	Asymmetric Digital Subscriber Line
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
APNS	Apple Push Notification Services
C2	CloudCell
CMX	Connected Mobile Experiences
CSR	Cloud Service Router
DAG	Directed Acyclic Graph
DSM	Distributed Shared Memory
DVFS	Dynamic Voltage and Frequency Scaling
FPS	Frames Per Second
GCM	Google Cloud Messaging
GPS	Global Positioning System
GRE	Generic Routing Encapsulation
HTTP	Hypertext Transfer Protocol
IaaS	Infrastructure as a Service
IDE	Integrated Development Environment
ILP	Integer Linear Programming
IPC	Inter-process communication
IPsec	Internet Protocol Security
JSON	JavaScript Object Notation
LAN	Local Area Network
MAN	Metropolitan Area Network

MCC	Mobile Cloud Computing
MI	Million Instructions
MIPS	Million Instructions Per Second
MPLS	Multiprotocol Label Switching
MPNS	Microsoft Push Notification Service
MSE	Mobility Services Engine
NVGRE	Network Virtualization using Generic Routing Encapsulation
OSGi	Open Service Gateway Initiative
PaaS	Platform as a Service
PM	Physical Machine
P2P	Peer-to-Peer
RDP	Remote Desktop Protocol
REE	Remote Execution Environment
REST	Representational State Transfer
RPC	Remote Procedure Call
SaaS	Software as a Service
SHA-1	Secure Hash Algorithm 1
SLA	Service Level Agreement
SL4A	Scripting Layer for Android
SLC	Seamless Cloud
SOA	Service-oriented Architecture
VCPU	Virtual CPU
VM	Virtual Machine
VMM	Virtual Machine Monitor
VNC	Virtual Network Computing
VPN	Virtual Private Network

VXLAN	Virtual Extensible LAN
WAN	Wide Area Network
WLAN	Wireless Local Area Network
XML	Extensible Markup Language
XMPP	Extensible Messaging and Presence Protocol

Contents

1	INTRODUCTION	17
1.1	Contextualization	18
1.2	Objectives	19
1.3	Thesis overview	20
1.4	Structure of the thesis	21
2	BACKGROUND	22
2.1	Mobile Cloud Computing	22
2.1.1	Types of Mobile Applications	23
2.1.2	Where to Perform Offloading?	24
2.1.3	Why to perform offloading?	27
2.1.4	When to perform offloading?	27
2.1.5	What to offload?	29
2.1.6	How to perform offloading?	30
2.1.7	Taxonomy	32
2.2	Programmable networks, SDN and NFV	32
2.2.1	Programmable seamless virtual clouds	34
2.3	Classification trees	37
2.3.1	General approach for solving a classification problem	37
2.3.2	Decision trees	39
2.4	Final considerations	43
3	STATE OF THE ART	44
3.1	Main offloading solutions	44
3.1.1	Discussion	48
3.2	Mobility support	50
3.3	MpOS	52
3.4	Final considerations	55
4	PROPOSED OFFLOADING SYSTEM	56
4.1	Architecture and components	57
4.1.1	Application Services	57
4.1.2	Offloading Dispatcher	58
4.1.3	REE Handler	59
4.1.4	Monitoring Service	60
4.1.5	Offloading Service	61

4.1.6	Offloading Decision Engine	61
4.1.7	OpenStack and Seamless Cloud Agent	62
4.1.8	Mobility Manager	63
4.2	Handling the offloading decision-making	64
4.2.1	System model	64
4.2.1.1	Model validation	66
4.2.2	Decision tree creation	68
4.2.3	Decision module operation and adaptive monitoring	69
4.3	Providing mobility support	70
4.3.1	Interconnecting geographically distributed remote servers	70
4.3.2	Tracking mobile devices and handling handovers	70
4.4	Final considerations	74
5	EVALUATION	76
5.1	Experiment 1 - Using decision trees in the decision module	76
5.1.1	Experiment 1.1 - The impact of the decision tree creation	76
5.1.2	Experiment 1.2 - The impact of the decision tree parsing	78
5.2	Experiment 2 - The performance of applications when offloading	78
5.2.1	Experiment 2.1 - BenchImage	80
5.2.2	Experiment 2.2 - MatrixOperations	83
5.2.3	Experiment 2.3 - CollisionBalls	85
5.3	Experiment 3 - Adaptive monitoring service evaluation	87
5.4	Experiment 4 - Mobility support	90
5.4.1	Experiment 4.1 - Vertical handover	90
5.4.2	Experiment 4.2 - Horizontal handover	92
5.5	Final considerations	95
6	CONCLUSION	97
6.1	Future work	99
	BIBLIOGRAPHY	101
	APPENDIX A – CLOUD-CELL TOPOLOGY AND CODE SAMPLES	108
	APPENDIX B – MCC VIRTUAL TESTBED	111