POLITECNICO DI MILANO

Scuola di Ingegneria dell'Informazione
Corso di Laurea Magistrale in Ingegneria Informatica
Dipartimento di Elettronica, Informazione e Bioingegneria

Generalized Nash Equilibria for the Service Provisioning Problem in Multi-Cloud Systems

Advisor: Prof. Danilo ARDAGNA
Co-Advisor: Prof. Mauro PASSACANTANDO

Master Thesis by:
Ettore TREVISIOL - 770628

Academic Year 2012/2013

This thesis is dedicated to my parents.
For their endless love, support and encouragement.

Acknowledgements

I would like to express my deepest gratitude to my advisor, Prof. Danilo Ardagna, for his excellent guidance, caring, patience and providing me with an excellent atmosphere for doing this thesis. My thanks go also to Prof. Mauro Passacantando for his great help in mathematical issues while developing this work.

I wish to thank my university mates, in particular thanks go to Giorgio Spadaro, Paolo Salvini and Riccardo Sacchi. I am also grateful to Davide Molinari, Anna Savi and the other students of the Software Engineering Laboratory for sharing their knowledge with me during the development of this thesis.

Finally, a particular thank goes to Andrea Fresco and Pier Paolo Cedaro, friends with whom I shared great moments of life.

Ettore

Contents

List of Figures

List of Tables

LIST OF TABLES

Abstract

In recent years the evolution and the widespread adoption of virtualization, service-oriented architectures, autonomic, and utility computing have converged letting a new paradigm to emerge: Cloud Computing. Clouds allow the on-demand delivering of software, hardware, and data as services. Currently the Cloud offer is becoming day by day wider, since all the major IT companies and Service providers, like Microsoft, Google, Amazon, HP, IBM, and VMware have started providing solutions involving this new technological paradigm.

As Cloud-based services are more numerous and dynamic, the development of efficient service provisioning policies becomes increasingly challenging. In this thesis we take the perspective of Software as a Service (SaaS) providers which host their applications at multiple Infrastructure as a Service (IaaS) providers. Each SaaS needs to comply with quality of service requirements, specified in Service Level Agreement (SLA) contracts with the end-users, which determine the revenues and penalties on the basis of the achieved performance level. Each SaaS provider wants to minimize the cost of use of Cloud resources and penalties for requests execution failures. Moreover, SaaS providers compete and bid for the use of infrastructural resources. On the other hand, the IaaSs want to maximize their revenues obtained providing virtualized resources.

In this thesis we model the service provisioning problem as a generalized Nash game. In particular, we develop an efficient distributed algorithm for the run-time allocation of IaaSs resources among competing SaaS providers. We demonstrate the effectiveness of our approach by performing tests considering realistic Cloud scenarios. Numerical results show that our algorithm is scalable and can be used at run-time since it can solve problem instances of maximum size in less than one minute. Compared to other state-of-the-art solutions our model can improve the efficiency of Cloud system evaluated in term of Price of Anarchy up to 100%. Furthermore our analyses point out the SaaS benefits while exploiting multiple IaaS deployment of applications and redistribution of application workloads.

Sommario

Negli ultimi anni l'evoluzione e la diffusa adozione di virtualizzazione, architetture orientate ai servizi, autonomic e utility computing sono confluiti in un nuovo paradigma: il Cloud Computing. Il Cloud Computing ha come obiettivo la fornitura on-demand di software e hardware come risorse accessibili tramite Internet.

Con l'aumento della quantità e della dinamicità dei servizi basati sul Cloud, lo sviluppo di politiche efficienti per la distribuzione delle risorse è diventato sempre più complesso. In questo lavoro di tesi abbiamo studiato il problema dal punto di vista dei fornitori Sofware as a Service (SaaS) che ospitano le loro applicazioni presso molteplici fornitori Infrastructure as a Service (IaaS). Ogni SaaS deve rispettare la qualità del servizio, specificata nei contratti di Service Level Agreement (SLA) con i propri clienti, che determina i ricavi e le penalità sulla base del livello di prestazioni raggiunto. Ogni SaaS vuole minimizzare i costi di utilizzo delle risorse Cloud e delle sanzioni causate dalla violazione dei contratti di SLA. Inoltre, i fornitori SaaS competono fra loro facendo offerte per l'utilizzo delle infrastrutture. Al contrario, gli IaaS vogliono massimizzare i propri introiti ottenuti fornendo risorse virtualizzate.

In questa tesi abbiamo modellato il problema per la fornitura delle risorse come un gioco di Nash generalizzato. In particolare, abbiamo sviluppato un algoritmo distribuito per la gestione a run-time delle risorse degli IaaS fra i SaaS in competizione. Abbiamo dimostrato l'efficacia del nostro approccio compiendo test rappresentativi di scenari di carico reale. I risultati numerici mostrano che l'algoritmo è scalabile e può essere utilizzato a run-time, poiché può risolvere istanze del problema di dimensione massima in meno di un minuto. Rispetto ad altre soluzioni della lettaratura il nostro modello può migliorare l'efficienza in termini di Price of Anarchy del sistema Cloud valutato fino al 100%. Inoltre, le nostre analisi evidenziano i benefici che i SaaS possono ottenere sfruttando il dislocamento delle applicazioni e la distribuzione del traffico su molteplici IaaS.

Chapter 1

Introduction

Cloud Computing has been a dominant IT news topic over the past few years. It is essentially a way for IT companies to deliver software/hardware on-demand as services through the Internet. Cloud Computing applications are generally priced on a subscription model, so end-users may pay a yearly usage fee, for example, rather than the more familiar model of purchasing software licenses. The Cloud-based services are not only restricted to software applications (Software as a Service – SaaS), but could also be the platform for the deployment and execution of applications developed in house (Platform as a Service – PaaS) and the hardware infrastructure (Infrastructure as a Service – IaaS).

In the SaaS paradigm, applications are available over the Web and provide Quality of Service (QoS) guarantees to end-users. The SaaS provider hosts both the application and the data, hence the end-user is able to use and access the service from all over the world. With PaaS, applications are developed and deployed on platforms transparently managed by the Cloud provider. The platform typically includes databases, middleware, and also development tools. In IaaS systems, virtual computer environments are provided as services and servers, storage, and network equipment can be outsourced by customers without the expertise to operate them.

Many companies, e.g., Google, Amazon, and Microsoft are offering Cloud Computing services such as Google's App Engine and Amazon's Elastic Compute Cloud (EC2) or Microsoft Windows Azure. Large data centers provide the infrastructure behind the Cloud and virtualization technology makes Cloud Computing resources more efficient and cost-effective both for providers and customers. Indeed, end-users obtain the benefits of the infrastructure without the need to implement and administer it directly adding or removing capacity almost instantaneously on a "pay-as-you-use" basis. Cloud providers can, on the other hand, maximize the utilization of their

physical resources also obtaining economies of scale.

The development of efficient service provisioning policies is among the major issues in Cloud research. Indeed, modern Clouds operate in a new and dynamic world, characterized by continuous changes in the environment and in the system and performance requirements must be satisfied. Continuous changes occurs without warning and in an unpredictable manner, and are outside the control of the Cloud provider. Therefore, advanced solution need to be developed to manage the Cloud system in a dynamically adaptive way, while continuously providing service and performance guarantees.

The recent evolution of Cloud system and the rapid growth of the Internet have led to a remarkable usage of game-theoretic tools. Problems arising in the ICT industry, such as resource or quality of service allocation problem, pricing, and load shedding, can not always be handled with classical optimization approaches because each player can be affected by the action of all players, not only by his own actions. In this context, game theory models and approaches allow to gain an in-depth analytical understanding of the service provisioning problem.

Game Theory has been successfully applied to diverse problems such as Internet pricing, flow and congestion control, routing, and networking. One of the most widely used "solution concept" in Game Theory is the Nash Equilibrium approach [71]: a set of strategies for the players constitute a Nash Equilibrium if no player can benefit by changing his/her strategy unilaterally or, in other words, every player is playing a *best response* to the strategy choices of his/her opponents.

In this thesis we take the perspective of SaaS providers which host their applications at multiple IaaS providers, thanks to a software layer developed within the MODAClouds project [68] [13]. Each SaaS provider wants to minimize the cost of use of Cloud resources and incurs in penalties in case of requests execution failures. The cost minimization is challenging since online services receive dynamic workloads that fluctuate during the day. Resources have to be allocated flexibly at run-time according to workload fluctuations. Furthermore, each SaaS behaves selfishly and competes with others SaaS for the use of infrastructural resources supplied by the IaaS. Each IaaS, in his turn, wants to maximize the revenues obtained providing the resources.

To capture the behavior of SaaSs and IaaSs in this conflicting situation in which the best choice for one depends on the choices of the others, we recur to the *Generalized Nash Equilibrium* (GNE) concept, which is an extension of the classical Nash equilibrium.

Therefore, the run-time service provisioning problem will be modeled as a *Generalized Nash Equilibrium Problem* (GNEP). We then use Noncooperative Game Theory results to develop an efficient algorithm for the run-time

management and allocation of IaaSs resources to competing SaaSs suitable also for a *fully distributed* implementation. Multiple solutions achieving generalized equilibria will be proposed and evaluated in terms of their efficiency with respect to the *social optimum* of the Cloud. We will demonstrate the effectiveness of our approach by simulation and performing tests on a real prototype environment.

The remainder of the thesis is organized as follows:

- In Chapter 2 we present a general overview on Cloud Computing providing definitions, illustrating the main characteristics and showing the different structures models available. Afterwards, we will explain the state of the art concepts and techniques relative to our work. An analysis and a classification of the literature approaches is given in terms of type of problem, solution found and discipline adopted, according to the approach used: pure optimization or game theory.
- In Chapter 3 the game-theoretic service provisioning problem will be faced. We will start introducing the problem statements and design assumptions. Then an analysis of the game and its properties will be performed. Finally, we will present an algorithm able to find an equilibrium for the resource allocation problem.
- In Chapter 4 we will describe the tools used in this thesis work. A description of the optimization modelers and solvers we adopted will be provided. Furthermore, we will describe the workload injector we have developed to estimate the performance parameters of Cloud applications.
- Chapter 5 will be dedicated to assess the quality of our solution through analyses and experiments. After a description of experiments settings, algorithm scalability and comparison with other two approaches will be performed. Finally, we will analyze the benefits that can be achieved by SaaS when hosting applications on multiple IaaSs.
- In Chapter 6 conclusions will be drawn, underling the achieved results and presenting future research directions.

Chapter 6

Conclusions

As Cloud-based services become more numerous and dynamic, resource provisioning becomes more and more challenging, especially when decisions can be affected by the action of others, not only by own actions, hence requiring game-theory approaches. Indeed, in any time instant resources have to be allocated to handle effectively workload fluctuations, while providing quality of service guarantees to the end users.

The overall goal we addressed in our thesis is the minimization of the costs associated with the allocated virtual machine instances in multiple IaaSs, while guaranteeing QoS constraints. To achieve this purpose we have proposed a game-theoretic approach for the run-time management of IaaSs provider capacities among multiple competing SaaSs. The cost model consists of objective functions which include revenues and penalties incurred depending on the achieved performance level and infrastructural costs associated with IaaSs resources. Therefore a distributed algorithm for identifying Generalized Nash Equilibria have been presented and its termination in a finite number of iterations has been demonstrated.

Thanks the AMPL language and the CPLEX solver, the effectiveness of our approach have been assessed by performing a wide set of analyses under multiple workload conditions. Realistic workloads created from a large website statistics and performance parameters estimated on an industrial benchmarking deployed in the Cloud have been used.

A number of different scenario of interest have been considered. Systems up to thousands of applications can be managed very efficiently in a fully distributed manner. Our algorithm found efficient GNE, for a hourly basis resource allocation, in less than a minute, proving to be perfectly suitable for run-time provisioning.

A comparison with utilization based state-of-the-art techniques and a rescaling algorithm shows that our solution outperform alternative methods

proving better results in terms of equilibrium efficiency both as regards the PoA both the IWC of each SaaS. In addition, our algorithm, achieve an efficient GNE under heavy workload conditions while thresholds based heuristic finds infeasible results due to the inability to manage SaaSs competition.

Finally analyses showed clear SaaS benefits while exploiting multiple IaaS deployment of applications and redistribution of traffic. SaaSs can have an average savings up to 50% compared to single IaaS architectures. The equilibria achieved are close to Cloud optimum, with inefficiency less than 2% for the social problem and of 6% for individual player in the worst cases.

Future work will extend the proposed solutions to consider multiple timescales for performing resource allocation from few minutes to one hour. Additionally short-term solutions will be based on receding horizon techniques.