Inf Syst Front
DOI 10.1007/s10796-012-9357-x

Cooperative game-based distributed resource allocation in horizontal dynamic cloud federation platform

Mohammad Mehedi Hassan · M. Shamim Hossain · A. M. Jehad Sarkar · Eui-Nam Huh

Abstract Distributed resource allocation is a very important and complex problem in emerging horizontal dynamic cloud federation (HDCF) platforms, where different cloud providers (CPs) collaborate dynamically to gain economies of scale and enlargements of their virtual machine (VM) infrastructure capabilities in order to meet consumer requirements. HDCF platforms differ from the existing vertical supply chain federation (VSCF) models in terms of establishing federation and dynamic pricing. There is a need to develop algorithms that can capture this complexity and easily solve distributed VM resource allocation problem in a HDCF platform. In this paper, we propose a cooperative game-theoretic solution that is mutually beneficial to the CPs. It is shown that in non-cooperative environment, the optimal aggregated benefit received by the CPs is not guaranteed. We study two utility maximizing cooperative resource allocation games in a HDCF environment. We use price-based resource allocation strategy and present both centralized and distributed algorithms to find optimal solutions to these games. Various simulations were carried out to verify the proposed algorithms. The simulation results demonstrate that the algorithms are effective, showing robust performance for resource allocation and requiring minimal computation time.

Keywords Horizontal dynamic cloud federation · Vertical supply chain federation · Distributed resource allocation · Cooperative games

M. M. Hassan (✉) · M. S. Hossain
Chair of Pervasive and Mobile Computing,
College of Computer and Information Sciences,
King Saud University, Riyadh, Saudi Arabia
e-mail: mmhassan@ksu.edu.sa

M. S. Hossain
e-mail: mshossain@ksu.edu.sa

M. M. Hassan · E.-N. Huh
Department of Computer Engineering,
Kyung Hee University, Seoul, South Korea

E.-N. Huh
e-mail: johnhuh@khu.ac.kr

A. M. J. Sarkar
Department of Digital Information Engineering,
Hankuk University of Foreign Studies,
Seoul, South Korea
e-mail: jehad@hufs.ac.kr

1 Introduction

In recent years, horizontal dynamic cloud federation (HDCF) models have emerged (Bittman 2008; Celesti et al. 2010a, b; Dodda et al. 2009), in which various CPs (smaller, medium, and large) collaborate dynamically to gain economies of scale and enlarge their virtual machine (VM) infrastructure capabilities (e.g. enlargement of Infrastructure-as-a-Service (IaaS) capability) to meet consumers' quality of service (QoS) targets, becoming themselves at the same time both 'users' and 'resource providers'. They differ from the existing vertical supply chain federation (VSCF) models (Rochwerger and Breitgand 2009; Ranjan and Buyya 2008; Buyya et al. 2010; Maximilien et al. 2009; Elmroth and Larsson 2009; OpenQRM 2010; di Costanzo et al. 2009; Costanzo et al. 2009; Assunção et al. 2010), in which CPs leverage cloud-based services of other clouds for seamless provisioning, and a priori agreements among the parties are needed to establish the federation. A HDCF platform dissolves as soon as the demand has been completed.

Previous research works (Bittman 2008; Celesti et al. 2010a, b; Dodda et al. 2009) have mainly focused on developing architecture, discovery, match-making, authentication and middleware solutions to build HDCF platforms among different CPs. One of the major research issues in an HDCF platform is the development of efficient distributed VM resource allocation policies, a topic that has gone unexplored in prior works. Distributed VM resource allocation is a challenge in an HDCF platform since the CPs (both resource users and providers) are inherently rational (i.e., self-interested and welfare-maximizing) due to their different ownerships and their dynamic, on-demand collaboration.

In an HDCF environment, there are two types of participants, a buyer CP, called a primary CP (pCP), and a seller or cooperating CP, known as a cCP. A CP can be simultaneously both a pCP and/or cCP. A pCP initiates an HDCF platform when it realizes that, at a certain time in the future, it will not be able to continue to provide services to its clients. Consequently, it transparently and dynamically enlarges its own virtualization infrastructure capabilities by asking for further VM resources from other cCPs for a specific period of time. Each pCP needs to comply with the QoS requirements specified in the Service Level Agreement (SLA) contracts with the clients. The goal is to maximize the pCP revenues from the SLAs while minimizing the cost of use of the VM resources supplied by the cCPs. Conversely, the cCPs want to maximize the revenues obtained from providing VM resources to the pCPs. Therefore, there is a need to develop an efficient mechanism for the economics of VM resource supply among CPs (pCPs and cCPs) with heterogeneous cost functions. Such a mechanism needs to be fair and ensure mutual benefits so that the participants are encouraged to join or form an HDCF platform.

To capture the inherently contradictory interests of the resource users (pCPs) and the resource providers (cCPs) in a HDCF platform, a natural modeling framework involves seeking an equilibrium or stable operating point for the system. In this context, game theoretic methods allow an in-depth, analytical understanding of the distributed VM resource-provisioning problem in an HDCF platform. Recently, game theory-based resource allocation mechanisms have received a considerable amount of attention in literature on cloud computing (Wei et al. 2010; Jalaparti et al. 2010; Lee et al. 2010; Teng and Magoulès 2010; Li et al. 2010). However, most of these works focus on optimal resource allocation using game theory in a single provider scenario. A few applicable studies have been found, focusing on developing effective VM resource allocations that differentiate the self-interested CPs and encourage them to contribute their VM resources to a HDCF platform.

In this paper, we analyze cooperative game theory-based VM resource allocation mechanisms for self-interested IaaS CPs that motivate them to participate in a HDCF platform. We show that, in a non-cooperative environment, the IaaS CPs (pCPs and cCPs) selfishly optimize their own utility. Thus the optimal aggregated benefit received by the IaaS CPs is not guaranteed, and the outcome may not meet expectations. In a HDCF platform, IaaS CPs (pCPs and cCPs) dynamically collaborate to complete tasks cooperatively on an on-demand basis, so that the performance requirements of all of the tasks can be satisfied without increasing the amount of VM resources. The contributions of this paper are as follows:

- We study cooperative game theoretic approaches to model the economics of VM resource supply among the IaaS CPs (pCPs and cCPs) in a HDCF environment. The games motivate the cooperation of different IaaS CPs with heterogeneous cost functions. We use price-based resource allocation strategies in these games to ensure proportional fairness of resource supplies.
- We propose two utility maximizing cooperative games- one for pCPs (UtilMaxpCP game) and the other for cCPs (UtilMaxcCP game), which led to two different optimal situations: a maximized total profit for pCPs and a maximized social welfare for cCPs. We also analyze a non-cooperative utility maximization game for cCPs (NonCopUtilMax-cCP) to show the necessity of cooperation for achieving individual optima and maximum social welfare in a HDCF platform.
- Both centralized and distributed algorithms are used in these games in order to achieve optimal solutions. These algorithms have low overhead and robust performance against dynamic pricing and stability. Various simulations were conducted to measure the effectiveness of these algorithms. The simulation results demonstrate that the UtilMaxcCP game is suitable in an HDCF platform as it is effective in terms of cost and enables the best set of cCPs to supply VM resources.

The paper is organized as follows: In Section 2, we describe the related literature regarding HDCF platforms and resource allocation mechanisms. In Section 3, we present the overall system architecture of a HDCF platform, and the mathematical problem formulation. In Section 4, we describe the two resource allocation games in detail. In Section 5, we evaluate the effectiveness of the proposed resource allocation games in a HDCF environment, and finally, in Section 6, we present our conclusions.

2 Related works

In this section, we provide an overview of the current cloud federation initiatives and present a comparative analysis of research related to horizontal dynamic cloud federation. We then discuss it in terms of the existing distributed resource allocation mechanisms in order to ascertain the feasibility of our proposed approaches.

2.1 Emergence of horizontal dynamic cloud federation

Cloud federation is gaining popularity in the research community, due to its flexibility and effectiveness in improving performance for end-users and achieving pervasive geographical coverage with increased capacity for a CP. Rochwerger and Breitgand (2009) first proposed an architecture for open federated cloud computing called the Reservoir model. The authors attempted to use grid interfaces and protocols to realize interoperability between the clouds or infrastructure providers; however, their work was in the model stage. Ranjan and Buyya (2008) described a decentralized overlay for a federation of enterprise Clouds called Aneka-Federation. Aneka is a .NET-based service-oriented resource management platform, which is based on the creation of containers that host the services. It is in charge of initializing services and act as a single point for interaction with the rest of the Aneka Cloud. Buyya et al. (2010) presented a vision and challenges of InterCloud for a utility-oriented federation of Cloud computing environments. IBM Altocumulus (Maximilien et al. 2009), a cloud middleware platform from IBM Almaden Services Research, aimed to solve the issue of managing applications across multiple clouds.

An analysis of the aforementioned works reveals that they focus on vertical supply chain federation (VSCF) Models, in which cloud providers leverage cloud services from other cloud providers for seamless provisioning (Celesti et al. 2010a, b; Bittman 2008). However, in the near future, we can expect that hundreds of cloud providers will compete to provide services to the thousands of users who want to run their complex heterogeneous applications on cloud computing environments. In these open Cloud collaboration scenarios, the existing VSCF models are not applicable. In fact, while clouds are typically heterogeneous and dynamic, the existing VSCF models are designed for static environments where a priori agreements among the parties are needed to establish the federation (Dodda et al. 2009; Celesti et al. 2010a).

Recent works like Bittman (2008), Celesti et al. (2010a, b) and Dodda et al. (2009) focus on horizontal collaborative Cloud service solutions, in which cloud providers (smaller, medium, and large) collaborate dynamically on an on-demand basis to gain economies of scale and enlarge their virtual infrastructure capabilities to meet the QoS targets of heterogeneous cloud service requirements. The author in Bittman (2008) first hypothesized the near future evolution of cloud computing in three stages: stage 1 'Monolithic' (now), cloud services are based on independent proprietary architectures; stage 2 'Vertical Supply Chain' (now) and stage 3 'Horizontal Federation'. In Celesti et al. (2010a), the authors proposed a HDCF solution based on a cross-cloud federation manager, a new component located within the cloud architectures, allowing a cloud to establish a federation with other clouds according to a three-phase model of discovery, match-making and authentication. An architecture for cross-cloud system management is proposed in Dodda et al. (2009), aimed at facilitating the management of computing resources from different cloud providers in an homogeneous manner. The primary goal is to provide flexibility and adaptability.

However, all of the aforementioned works only provide frameworks and policies required to achieve a HDCF model among CPs, excluding the decision making mechanisms for allocating VM resources to collaborators and the strategies for motivating and forming federations. VM resource allocation is challenging in a HDCF platform, since the CPs (both resource users and suppliers) are inherently rational (i.e. self-interested and welfare- maximizing) due to their different ownerships and collaborate dynamically on an on-demand basis. Rational CPs make their own decisions according to their budgets, capabilities, goals and local knowledge, without considering the global good. Furthermore, there is no omniscient designer who can develop a resource allocation mechanism that satisfies the preferences of all rational resource users and suppliers while maximizing the global efficiency. Global efficiency is generated through interactions among agents from the bottom up. Thus, we address this issue and focus on developing an efficient mechanism to model the economics of VM resource supply among CPs (pCPs and cCPs) with heterogeneous cost functions.

2.2 Distributed resource allocation mechanisms for a HDCF platform

In a HDCF environment, distributed VM resource allocation mechanisms should be established from the bottom up, meaning that every rational CP (resource user/ pCP or supplier/cCP) makes individual decisions based on local knowledge and preferences, without considering the global good. The global efficiency is generated from the bottom up through interactions among CPs (pCP and cCPs). Each CP can be affected by the actions of all CPs, not only her own action. In this setting, a natural modeling framework involves seeking an equilibrium, or a stable operating point, for the system.

Economic-based approaches like commodity market-based or auction-based resource allocation mechanisms (Gomes et al. 2010; An et al. 2010; Auyoung et al. 2004; Fu et al. 2003; Lai et al. 2005; Macias and Guitart 2010) can be useful to build a distributed VM resource allocation mechanism in a HDCF platform. For example, the authors in Gomes et al. (2010) investigate the application of market oriented mechanisms based on the General Equilibrium Theory of Microeconomics to coordinate the sharing of resources between the clouds in the federated cloud. However, the most obvious weakness of the commodity market-based/auction-based mechanism for resource allocation in a HDCF platform is that no such real market exists. The difficulty lies in verifying that the empirical results in the experimental settings can be duplicated in real markets (Wolski et al. 2003). Also in such a market setting we have to ensure truth-elicitation of CPs since the economic incentives tends to induce self-interested and welfare-maximizing providers to alter their bids or prices in order to increase their revenue (He and Ioerger 2005). In these contexts, game theory based market-model for resource allocation is the most preferable solution. It allows for an in-depth analytical understanding of the distributed VM resource provisioning problem in a HDCF platform.

Game-theory based distributed resource allocation mechanisms have received a considerable amount of attention in different areas, such as grid computing (He and Ioerger 2005; Carroll and Grosu 2010; Subrata and Zomaya 2008; Andrade et al. 2007; Wilkins et al. 2010; Penmatsa and Chronopoulos 2011; Cheng et al. 2010; Khan and Ahmad 2006), P2P network (Ma et al. 2006; Kumar et al. 2011; Park and van der Schaar 2010), and recently in Cloud computing area (Ardagna et al. 2011; Wei et al. 2010; Jalaparti et al. 2010; Lee et al. 2010; Teng and Magoulès 2010; Li et al. 2010; Antoniadis et al. 2010; An et al. 2010). In cloud computing, researchers have (Wei et al. 2010) proposed a game-theoretic method for scheduling cloud-based computing services with collaborative QoS requirements. In Jalaparti et al. (2010), the authors modeled the complex client-client and client-provider interactions in a cloud using game theory. A new game theory based resource allocation using the Bayesian Nash Equilibrium was proposed in Teng and Magoulès (2010). In Li et al. (2010), the authors addressed the problem of optimal resource allocation in computational cloud. An evolutionary game theoretical mechanism for adaptive and stable application deployment in clouds was proposed in Lee et al. (2010). In Ardagna et al. (2011) a game theory based approach for the run time management of a IaaS provider capacity among multiple competing Software-as-a-Services (SaaSs) is proposed.

However, a little research is directly applicable in an HDCF scenario. In this context, a federation of cloud providers was analyzed to try to determine the profit share with the notion of diversity using a coalitional game theory in Antoniadis et al. (2010). However, the researchers assumed a static federation of cloud providers and thus did not address the issues of dynamic and distributed coalition formation, nor did they include the motivations that would encourage CPs to join or form a horizontal cloud federation.

In An et al. (2010), the authors proposed a market-based resource allocation approach where consumers (buyers) are dynamic and can bid for a set of cloud resources from multiple cloud providers (sellers). Instead of using auction-based approaches, they proposed a distributed negotiation mechanism where agents (buyers and sellers) negotiate over both a contract price and a decommitment penalty, which allows agents to decommit from contracts at a cost. Although the work seems to be applicable in an HDCF platform, it has some major drawbacks. First, it is different from an HDCF platform, since it considers a combinatorial market where consumers/users can get multiple cloud resources from multiple providers. Second, the resource allocation model assumes that each agent (buyer and seller) can know the demand/supply ratio of each resource. In reality, an agent may not know the demand/supply ratio. Fourth, the proposed approach also assumes that each buyer agent knows each seller's expected cost of providing a resource. In a distributed dynamic environment, this is impractical. Finally the proposed approach does not investigate the agent's rational strategies and equilibrium criteria, which are important in dynamic resource allocation scenarios.

He and Ioerger (2005) showed that providing incentives for agents to share their resources with others is the key to making computational grids realistic. However, they did not evaluate the social welfare of a coalition formation-based resource allocation mechanism, which reflects the level of satisfaction of the participants in the coalition. Carroll and Grosu (2010) proposed a coalition game theory-based resource composition framework among self-interested grid service providers for creating virtual organizations in Grids. The authors tried to compute the worth of each coalition for an agent that maximizes its profit. However, a service provider cannot possibly compute the worth of all coalitions and, thus, has limited information on which to base its decision. Andrade et al. (2007) proposed an incentive mechanism based on the local history in P2P grids, which makes it in the interest of each participating peer to contribute its spare resources. It is well-known that incentive mechanisms using local history are not effective in all settings (Feldman et al. 2004). In Cheng et al. (2010), a novel approach for enabling grid users to perform resource federation using intelligent agent negotiation is proposed. The features of the agents were autonomous and distributed and were designed to address the federation problems in grids, such as resource selection and policy reconciliation. Khan and Ahmad (2006) proposed non-cooperative, semi-cooperative and cooperative games of resource allocation for computational grid environments. However, their cooperative method has a high computational complexity and is difficult to implement.

The main differences between the aforementioned related works and our work are as follows:

1. Firstly, we address the problem of finding effective VM resource allocation mechanisms that differentiate the self-interested CPs and motivate them to contribute their VM resource to a HDCF platform. To the best of our knowledge this is the first paper analyzing VM resource allocation in a HDCF platform.
2. Secondly, we motivate the providers to supply VM resource based on a defined price function, which ensures proportional fairness of resource supply.
3. Thirdly, we propose two distributed cooperative algorithms for implementing the games. These have low overhead and robust performance against stability and dynamic pricing. We also analyze a non-cooperative game to show the effectiveness of cooperation.
4. Finally, a cost effective resource allocation game is achieved among IaaS CPs which encourages them to form an HDCF platform.

3 System model and problem formulation

In this section, we first present the overall system architecture of a HDCF platform. Then, we describe our mathematical problem formulation of the resource allocation games.

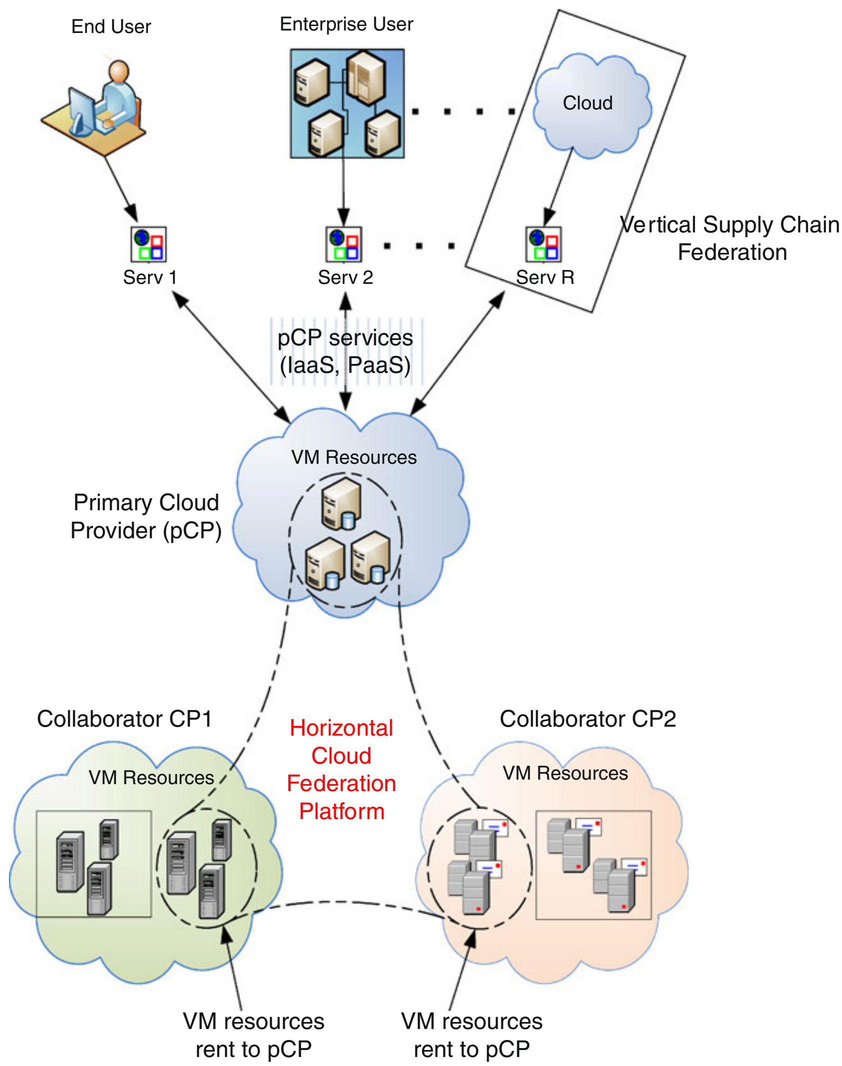
3.1 Overview of horizontal dynamic cloud federation platform

Let us provide an overview of a HDCF platform as described in Celesti et al. (2010a, b). We assume that CPs are rational (self-interested and well fare maximizing) and make their own decisions according to their budgets, capabilities, goals and local knowledge. The formation of an HDCF platform is initiated by a IaaS CP, known as a primary cloud provider (pCP), when it realizes that at a certain time in the future it cannot continue providing services to its clients (i.e. other clouds, enterprises, generic end users, etc). Consequently, it transparently and dynamically enlarges its own virtualization infrastructure capabilities by asking for further VM resources from other collaborating clouds, called cCPs, for a specific period of time. A IaaS cloud provider could be simultaneously both a pCP and/or a cCP.

The Fig. 1 shows a formed horizontal dynamic cloud federation platform. We can see that the pCP is already in VSCF model because it is able to provide services to other clouds (top part of the Fig. 1). Moreover it is also dynamically collaborating with other CPs, that is, cCP's, to enlarge its capabilities when it realizes that its virtualization infrastructure would be unable to continue providing services to its clients. Thus, a HDCF platform allows IaaS CPs to cooperatively achieve greater scales and reaches, as well as service qualities and performances, than could otherwise be attained individually. Its significance can be better understood through the following two example applications:

– Emerging Cloud applications like Social networks (e.g. Facebook, MyS- pace etc.) deployed on a CP serve dynamic content to millions of users, whose access and interaction patterns are hard to predict. In addition, the dynamic creation of new plug-ins by independent developers may require additional resources which may not be provided by the hosting cloud provider at certain periods in time. In this

Fig. 1 A formed horizontal dynamic cloud federation Platform



situation, load spikes (cloud bursting) can occur at different locations at any time, for instance, whenever new system features become popular or a new plug-in application is deployed. This results in an SLA violation and ends up incurring additional costs for the CP (Buyya et al. 2010). This necessitates building mechanisms for horizontal dynamic collaborations of CPs for seamless provisioning of VM resources.

– Other example applications that need horizontal dynamic cloud federation are massively multi-player online role-playing games (MMORPGs). World of Warcraft (http://www.worldofwarcraft.com/cataclysm/), for example, currently has 11.5 million subscribers; each of whom designs an avatar and interacts with other subscribers in an online universe. Second Life (http://secondlife.com/) is an even more interesting example of a social space that can be created through dynamic Cloud collaboration. Any of the 15 million users can build virtual objects, create their own virtual land, buy and sell virtual goods, attend virtual concerts, bars, weddings, and churches, and communicate with any other member of the virtual world. These MMORPGs require huge amount of Cloud resources/services which cannot be provided by a single cloud provider at that time. This necessitates building mechanisms for the seamless collaboration of different CPs supporting dynamic scaling of resources across multiple domains in order to meet the QoS targets of MMORPGs customers.

However, an effective VM resource allocation that differentiates the self- interested IaaS CPs and motivates them to contribute their VM resources to a HDCF platform is a complex issue, as is how to decide the resources that each cCP should contribute to the platform and the resource that each pCP can utilize. In game theory, coalition/federation formation is a cooperative game among self- interested agents. Therefore, we studied different cooperative game theory-based distributed VM resource allocation mechanisms for IaaS CPs in HDCF platforms.

From the pCPs perspective, VM resources should continue providing services to their clients at lower operating systems costs. Thus, they define various price functions, which specify how much price per hour should be given to cCPs for each unit of the VM resource supplied in the HDCF platform. From the cCPs perspective, the goal is to maximize their utilities by selling the VM resources based on the price functions of the pCPs. A IaaS CP (pCP or cCP) will not join the platform if its net profit is zero. In this paper, we analyzed the interactions among IaaS CPs (pCPs and cCps) under different resource allocation games.

3.2 Mathematical model formulation in a HDCF platform

The notations used in the paper are summarized in Table 1. A pCP requires VM resources with specific QoS requirements during a certain period t to continue providing services to its clients. A set of cCPs $P = \{P_i^t | i = 1 \ldots m\}$ is available during that period which can form a HDCF platform with pCP by providing VM resources with the required QoS. Let R_{VM}^t be the total VM resources supplied in a HDCF platform for period t, r_i^t be the units of VM resource supplied by a CP i for period t, and $\tilde{C}_i^t$ be its maximum capacity during that period. The sum of the VM resources supplied to any pCP should be $\sum_{i=1}^{m} r_i^t = R_{VM}^t$. We know that the pCP can buy these VMs cheaper than the revenue it obtains from selling them to clients (Goiri et al. 2010). The definitions used for the mathematical model formulation are as follows:

Definition 1 (Profit) Let $\mathrm{Rev}_{cCP}^t(R_{VM}^t)$ be the revenue a pCP can provide for getting R_{VM}^t resources from cCPs at certain period t and $\mathrm{Pr}_{cCP}^t(R_{VM}^t)$ be the price per hour set by a pCP to cCPs for each unit of VM resource supplied in period t. Then, the expected profit of a pCP obtained from executing tasks on R_{VM}^t resources from cCPs is defined as follows:

$$\mathrm{Profit}_{pCP}^t\left(R_{VM}^t\right)$$
$$= \mathrm{Rev}_{cCP}^t\left(R_{VM}^t\right) - R_{VM}^t \cdot \underset{cCP}{\overset{t}{\mathrm{Pr}}}\left(R_{VM}^t\right) \quad (1)$$

As shown in Eq. 1, the total profit is determined by the total VM resource R_{VM}^t supplied in the HDCF platform and a pCP can only influence the value of R_{VM}^t by setting a proper price function $\mathrm{Pr}_{cCP}^t(R_{VM}^t)$. Threfore, the pCP can strategically define the price function $\mathrm{Pr}_{cCP}^t(R_{VM}^t)$ in such a way that it can motivate the available cCPs to contribute their resources to a HDCF platform and get profit from this transaction.

Definition 2 (Cost function) Let M_i^t be the production cost of the first unit of VM resource for any provider i during a certain period t and α_i is its learning factor. Then, in order to supply r_i^t units of the VM resource, any cCP i has to pay $Cost(r_i{}^t)$, which is defined as follows (Amit and Xia 2011):

$$Cost\left(r_i^t\right) = \frac{M_i^t \cdot {r_i^t}^{1+\log_2\alpha}}{1+\log_2\alpha} \quad (2)$$

$$s.t. \quad 0 \leq r_i^t \leq \tilde{C}_i^t \quad (3)$$

Table 1 Summary of notations

Parameters	Description
R_{VM}^t	Total VM resources supplied in a HDCF platform in period t
$P = \left\{P_i^t \mid i = 1 \ldots m\right\}$	Total number of cloud providers present in period t
r_i^t	VM resource supplied by provider i in period t
$\tilde{C}_i^t$	Total VM capacity of provider i in period t
$Cost\left(r_i^t\right)$	Cost of supplying r_i^t unit of VM resource by provider i in period t
M_i^t	Cost of the first unit of VM resource by provider i in period t
α_i	Learning factor of provider i where $0.75 < \alpha_i^t < 0.9$
ω	Parameter defining the rate of revenue in a HDCF platform
$\mathrm{Profit}_{pCP}^t\left(R_{VM}^t\right)$	Profit of any pCP with R_{VM}^t resources supplied from cCPs in period t
$\mathrm{Rev}_{cCP}^t\left(R_{VM}^t\right)$	Revenue function estimated by a pCP for R_{VM}^t resources supplied by cCPs in period t
$\mathrm{Pr}_{cCP}^t\left(R_{VM}^t\right)$	Price per hour given to cCPs by a pCP for each unit of VM resource supplied in period t
$Util\left(r_i^t\right)$	Utility of any cCP i by providing r_i^t unit of VM resources in period t

This cost function is based on the learning curve model (Amit and Xia 2011) which assumes that as the number of production units are doubled the marginal cost of production decreases by a fixed factor α. The marginal cost function of Eq. 2 is as follows:

$$MCost\left(r_i^t\right) = M_i^t \cdot r_i^{t^{\log_2 \alpha}} \tag{4}$$

The cost function can be heterogeneous for different cCPs based on α and M. The higher the value of α and M, the higher the production cost for the provider. It has been reported (Amit and Xia 2011) that for a typical CP, as the total number of servers in the house doubles, the marginal cost of deploying and maintaining each server decreases 10–25%, thus the learning factors are typically within the range (0.75, 0.9).

Definition 3 (Revenue function) Let ω be the increasing rate of revenue. A pCP can estimate the revenue function of the HDCF platform as follows:

$$\mathrm{Re}v_{cCP}^t\left(R_{VM}^t\right) = \frac{M \cdot \left(1 - e^{-R_{VM}^t \cdot \omega}\right)}{\omega} \tag{5}$$

The function $\mathrm{Re}v_{cCP}^t(R_{VM}^t)$ is a non-decreasing and concave function which means that the more resources supplied by cCPs, the higher the revenue. However, the marginal revenue decrease as the resource increase. The product of the price and the total available VM unit R_{VM}^t should not exceed the corresponding revenue. The pCP has the freedom to decide how much revenue is to be provided to the cCPs by varying the parameters M and ω.

Definition 4 (Price function) Based on the revenue function of Eq. 5, the price per hour given to the cCPs by a pCP for each unit of VM resource supplied in period t is defined as follows:

$$\underset{cCP}{\overset{t}{\Pr}}\left(R_{VM}^t\right) = M \cdot e^{-R_{VM}^t \cdot \omega} \tag{6}$$

$$s.t. \quad R_{VM}^t > 0 \tag{7}$$

The function $\Pr_{cCP}^t(R_{VM}^t)$ is the marginal gain of the HDCF platform. When the amount of VM resource increase, the price per hour of each unit of VM resource decrease. Also this function represents the *proportional fairness* of contributing resources by cCPs.

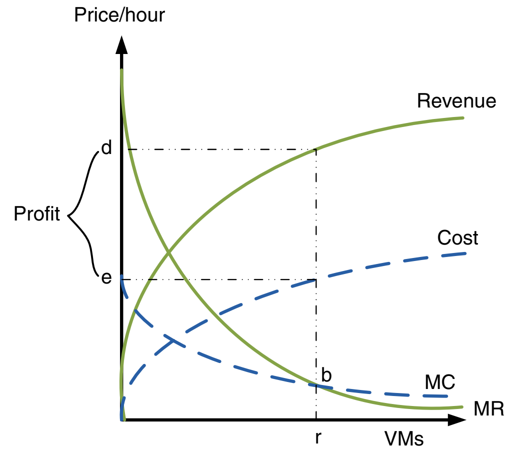


Fig. 2 Maximum profit using marginal revenue and cost

Definition 5 (Utility of any cCP) The utility of any cCP i which represent its level of satisfaction of supplying r_i^t units of VM resources is defined as follows:

$$Util(r_i^t) = r_i^t \cdot \underset{cCP}{\overset{t}{\Pr}}\left(R_{VM}^t\right) - Cost\left(r_i^t\right) \tag{8}$$

$$s.t. \quad 0 \le r_i^t \le \tilde{C}_i^t \tag{9}$$

All cCPs are rational, and they strategically choose the amount of r_i^t so as to maximize their profit. A cCP can get maximum profit when the marginal costs (MC) equal the marginal price (MR). Figure 2 shows the MC and MR of our model based on Eqs. 4 and 6. The profit or utility is maximized when $MC = MR$ at point b, thus r VMs. Beyond r, each VM costs more than the revenue offered ($MC > MR$). Until r VMs is reached, the cost for each unit of VM is less than the revenue obtained from it ($MC < MR$). For r VMs, the total cost and revenue are e and d respectively; so the maximum profit is $d - e$.

4 Resource allocation games in a HDCF platform

The objective of a resource allocation game in a HDCF platform is to dynamically allocate VM resources among the IaaS CPs (pCPs and cCPs) on an on-demand basis without exceeding their resource capacities and expense prices, thus satisfying all involved. We study two cooperative resource allocation games in a HDCF platform. Both games are repeated and asynchronous. In both games, a pCP strategically defines a price function $\Pr_{cCP}^t(R_{VM}^t)$ and publicizes it along with the total amount of VM resources supplied. Each cCP only knows this information and can update its own strategy during each move so as to maximize its utility.

In one game, pCPs, acting as leaders, tried to maximize their utilities, with cCPs also trying to maximize their utilities in a non-cooperative manner. In another game, pCPs just announced their prices and did not attempt to maximize their profit or utilities. The cCPs in this game cooperatively tried to maximize their social welfare or utilities, by deciding collectively the amount of VM resources to supply based on the prices of pCPs.

4.1 Utility maximization game for pCPs (UtilMaxpCP)

In this cooperative game of price-based resource allocation, a pCP, acting as a Stackelberg leader (Drew and Jean 1993), strategically decides a price function and the cCPs react by selecting the ideal amount of VM resources to supply. The objective of this game is to find the Stackelberg Equilibrium (SE), from which neither the leader (pCP) nor the followers (cCPs) have incentives to deviate. A pCP tries to set a proper price so as to maximize its profit, whereas the cCPs try to maximize their benefits based on that price. In this game, cooperation means that the cCPs are following the pCP or the leader to reach a SE. The formulation of this game in an HDCF platform can be described as follows:

1. **Players:** There are two types: the pCP as a leader and the available cCPs as followers.
2. **Strategies:** The strategy for a pCP is to maximize its $Profit^t_{pCP}(R^t_{VM})$ in the HDCF platform by defining a price function $\mathrm{Pr}^t_{cCP}(R^t_{VM})$. However, for a defined price function $\mathrm{Pr}^t_{cCP}(R^t_{VM})$, there is no guarantee that a pCP will maximize its total profit. Thus, a pCP can set an initial constant or uniform price and can choose a proper price Pr^t_{cCP} to maximize its total profit. The strategy for the cCPs are to maximize their benefits based on r^t_i and the price Pr^t_{cCP}. They play a non-cooperative game with each other to maximize their own utilities.
3. **Payoffs:** The payoff for a pCP is the utility it can gain by using VM resources for a minimal amount from the available cCPs. For a cCP it is the utility gained from selling its own VM resources to a pCP.

For the aforementioned game model, the objective of the pCP is to maximize its profit/utility by finding a proper price Pr^t_{cCP}. This is obtained by solving the following optimization problem:

$$Max\ \mathrm{Profit}^t_{pCP}\left(R^t_{VM}\right) = \mathrm{Rev}^t_{cCP}\left(R^t_{VM}\right) - R^t_{VM}\cdot \underset{cCP}{\overset{t}{\mathrm{Pr}}} \tag{10}$$

$$s.t.\quad \underset{cCP}{\overset{t}{\mathrm{Pr}}} \geq 0 \tag{11}$$

$$0 \leq r^t_i \leq \tilde{C}^t_i \tag{12}$$

The objective of each cCP i is to selfishly maximize its profit/utility based on the price Pr^t_{cCP} from a pCP. Mathematically, for each cCP i, this problem can be formulated as:

$$Max\ Util\left(r^t_i\right) = r^t_i \cdot Pr^t_{cCP} - Cost\left(r^t_i\right) \tag{13}$$

$$s.t.\quad 0 \leq r^t_i \leq \tilde{C}^t_i \tag{14}$$

The problems in Eqs. 10 and 13 form a Stackelberg game. Generally, he SE for a Stackelberg game can be obtained by finding its sub game perfect Nash Equilibrium (NE). In the proposed game, it was not difficult to see that the cCPs strictly competed in a non-cooperative fashion. Therefore, a non- cooperative resource allocation sub game was formulated for the cCPs. For a non-cooperative game, NE is defined as the operating point(s) at which no player can improve utility by changing its strategy unilaterally, assuming everyone else continues to use their current strategy. For the pCPs, since there is only one player, the best response of the pCP is to solve the problem in Eq. 10. To achieve this, the best response functions for the followers (cCPs) must be obtained first by solving Eq. 13, since the leader (pCP) derives its best response function based on those of the followers or cCPs.

The problem in Eq. 13 is a concave function over Pr^t_{cCP} , and the boundary constraint $0 \leq r^t_i \leq \tilde{C}^t_i$ is affine. Thus, the optimal solution must satisfy the Karush–Kuhn–Tucker (KKT) conditions. Therefore, by solving the KKT conditions, the optimal solution for the problem in Eq. 13 can be easily obtained in the following lemma.

Lemma 1 *For a given price Pr^t_{cCP}, the optimal solution for the problem in Eq. 13 is given by*

$$r^{t*}_i = \begin{cases} \left[\dfrac{M^t_i}{Pr^t_{cCP}}\right]^{-\frac{1}{\log_2 \alpha_i}}, & 0 \leq r^t_i \leq \tilde{C}^t_i, \\ \tilde{C}^t_i, & r^t_i > \tilde{C}^t_i, \\ 0, & r^t_i \leq 0 \end{cases} \tag{15}$$

Proof Since the problem in Eq. 13 has a boundary constraint for the variable r^t_i, that is, $0 \leq r^t_i \leq \tilde{C}^t_i$, it can be formulated as a constrained optimization, which can be solved by the method of Lagrangian Multiplier.

$$L = Util\left(r_i^t\right) - \sum_{i=1}^{m} \gamma r_i^t + \sum_{i=1}^{m} \varphi\left(r_i^t - \tilde{C}_i^t\right) \tag{16}$$

where γ and φ are the Lagrangian constant. The Karush Kuhn Tucker (KKT) condition is as follows:

$$\frac{\partial L}{\partial r_i^t} = Util'\left(r_i^t\right) - \gamma + \varphi = 0, \quad i = 1, \ldots\ldots, m \tag{17}$$

$$\gamma \geqslant 0,\ \varphi \geqslant 0,\ \gamma r_i^t = 0,\ \varphi\left(r_i^t - \tilde{C}_i^t\right) = 0,$$
$$0 \leq r_i^t \leq \tilde{C}_i^t,\ i = 1, \ldots\ldots, m \tag{18}$$

$$Util'\left(r_i^t\right) = M_i^t \cdot r_i^{t\log_2 \alpha_i} = Pr_{cCP}^t \tag{19}$$

$$r_i^t = \left[\frac{M_i^t}{Pr_{cCP}^t}\right]^{-\frac{1}{\log_2 \alpha_i}} \tag{20}$$

By solving r_i^t of Eq. 19, the optimal r_i^{t*} of each cCP i is either r_i^t if the maximum located in the range of $(0, \tilde{C}_i^t)$, or the boundary value 0 or $\tilde{C}_i^t$. Lemma 1 is thus proved. □

From Lemma 1, it is observed that for some cCPs, the r_i^{t*} value can be less than or equal to 0. So those cCPs will be removed from the game. So, given the value of proper price Pr_{cCP}^t, a pCP can predict the total VM resource R_{VM}^{t*} contributed to the system, that is

$$R_{VM}^{t*} = \sum_{i=1}^{m} r_i^{t*} \tag{21}$$

Now, let's consider the optimization problem of a pCP in Eq. 10. If the pCP knows the parameters M and α of all the cCPs, it can formulate its own maximization, which aims at maximizing the total profit with respect to R_{VM}^{t*}

$$Max\ \mathrm{Profit}_{pCP}^t\left(R_{VM}^{t*}\right) = \mathrm{Rev}_{cCP}^t\left(R_{VM}^{t*}\right) - R_{VM}^{t*} \cdot \underset{cCP}{\overset{t}{\mathrm{Pr}}} \tag{22}$$

$$s.t.\quad \underset{cCP}{\overset{t}{\mathrm{Pr}}} \geq 0 \tag{23}$$

$$0 \leq r_i^t \leq \tilde{C}_i^t \tag{24}$$

Since the total amount of VM resources R_{VM}^{t*} solely depends on the value of price Pr_{cCP}^t through Eqs. 15 and 21, the objective function can be rewritten by substituting R_{VM}^{t*} in terms of Pr_{cCP}^t as follows:

$$Max\ \mathrm{Pr}ofit_{pCP}^t\left(\underset{cCP}{\overset{t}{\mathrm{Pr}}}\right)$$
$$= \mathrm{Rev}_{cCP}^t\left(\sum_{i=1}^{m}\left[\frac{M_i^t}{\mathrm{Pr}_{cCP}^t}\right]^{-\frac{1}{\log_2 \alpha_i}}\right)$$
$$-\left(\sum_{i=1}^{m}\left[\frac{M_i^t}{\mathrm{Pr}_{cCP}^t}\right]^{-\frac{1}{\log_2 \alpha_i}}\right) \cdot \underset{cCP}{\overset{t}{\mathrm{Pr}}} \tag{25}$$

$$= \frac{M \cdot \left(1 - e^{-\sum_{i=1}^{m}\left[\frac{M_i^t}{\mathrm{Pr}_{cCP}^t}\right]^{-\frac{1}{\log_2 \alpha_i}} \cdot \omega}\right)}{\omega}$$
$$-\left(\sum_{i=1}^{m}\left[\frac{M_i^t}{\mathrm{Pr}_{cCP}^t}\right]^{-\frac{1}{\log_2 \alpha_i}}\right) \cdot \underset{cCP}{\overset{t}{\mathrm{Pr}}} \tag{26}$$

$$s.t.\quad 0 \leq r_i^t \leq \tilde{C}_i^t \tag{27}$$

Although it is difficult to find a close-form solution of Pr_{cCP}^{t*} for Eq. 26, we can solve this optimization efficiently without derivative using a direct search method, for example, pattern search method as described in Kolda et al. (2003) was applied to solve the value of the optimal price Pr_{cCP}^{t*}. Once the pCP finds the optimal price, it can calculate the value of all r_i^{t*} using Eq. 15. However, the boundary constraints in Eq. 27 may be violated, and in which case the problem becomes more complicated. Still the solution can be found using Lagrangian multiplier. Without the constraints, the objective function in Eq. 26 is a concave function. Hence, there exists a unique solution that satisfies the KKT-conditions of Eq. 26 as follows:

$$L = \mathrm{Pr}ofit_{pCP}^t\left(\underset{cCP}{\overset{t}{\mathrm{Pr}}}\right) - \sum_{i=1}^{m} \gamma r_i^t + \sum_{i=1}^{m} \varphi\left(r_i^t - \tilde{C}_i^t\right) \tag{28}$$

$$\frac{\partial L}{\partial\, \mathrm{Pr}_{cCP}^t} = \frac{\partial\left[\mathrm{Pr}ofit_{pCP}^t\left(\mathrm{Pr}_{cCP}^t\right)\right]}{\partial\, \mathrm{Pr}_{cCP}^t} - \frac{\partial\left[\sum_{i=1}^{m} \gamma r_i^t\right]}{\partial\, \mathrm{Pr}_{cCP}^t}$$
$$+ \frac{\partial\left[\sum_{i=1}^{m} \varphi\left(r_i^t - \tilde{C}_i^t\right)\right]}{\partial\, \mathrm{Pr}_{cCP}^t} = 0 \tag{29}$$

$$\gamma \geqslant 0,\quad \varphi \geqslant 0,\ \gamma r_i^t = 0,\ \varphi\left(r_i^t - \tilde{C}_i^t\right) = 0,$$
$$0 \leqslant r_i^t \leqslant \tilde{C}_i^t,\ i = 1, \ldots\ldots, m \tag{30}$$

From the KKT conditions, if $\gamma = 0$, $and\ \varphi = 0$, then all the r_i^t lie between $[0, \tilde{C}_i^t]$. When the boundary constraints are violated ($\gamma \neq 0$, $or\ \varphi \neq 0$), the value of r_i^t are forced to be the boundary value [either $0\ or\ \tilde{C}_i^t$]. If r_i^t is less than or equal to zero for a certain cCP i, this cCP is not eligible for contributing as the cost of supplying the VM is comparatively high. Similarly, if r_i^t is greater than $\tilde{C}_i^t$, the cCP has optimal value of r_i^t. This cCP should provide as much VM resource as possible since the cost is comparatively low. Thus, some cCPs can be eliminated, whose values of r_i^t are already known from the problem formulation and we can resolve the r_i^t for the remaining cCPs.

Until now, we assume that the pCP knows the characteristics of the cost function of each cCP such that it can determine the behavior of the cCPs, and it can construct its own objective function. However, in distributed environment, the pCP can only observe the action of each cCP by setting a probing price. The cCPs choose the best r_i^t to maximize their net utility. The pCP keeps adjusting the price gradually until a desirable profit is obtained. Now we present a distributed algorithm to find the optimal value of Pr_{cCP}^t.

The algorithm is described step by step as follows:

Step 1: Initialize the probing price $\mathrm{Pr}_{cCP}^t = 0.1$ and $\{r_i^t\}_{i=1}^P = 0$.

Step 2: Send $\mathrm{Pr}_{cCP}^t = 0.1$ to all cCPs and receive corresponding $R_{VM}^t = \sum_{i=1}^P r_i^t$.

Step 3: if $\mathrm{Profit}_{pCP}^t(R_{VM}^{t*}) = \mathrm{Rev}_{cCP}^t(R_{VM}^{t*}) - R_{VM}^{t*} \cdot \underset{cCP}{\overset{t}{\mathrm{Pr}}}$ is maximized or $\frac{\partial\left[\mathrm{Pr}\,ofit_{pCP}^t(\mathrm{Pr}_{cCP}^t)\right]}{\partial\,\mathrm{Pr}_{cCP}^t} = 0$, the optimal Pr_{cCP}^t is found and break. Otherwise update Pr_{cCP}^t based on old price and the percentage change of the net profit.

Step 4: If $0 \leq r_i^t \leq \tilde{C}_i^t$ for all $i \in P$, then break.

Step 5: Now for some cCP, $i \in P$, If $r_i^t \leq 0$, remove those cCPs from the list of P. Also for some cCP, $i \in P$, If $r_i^t \geq \tilde{C}_i^t$, set $r_i^t = \tilde{C}_i^t$.

The aforementioned search method is a zero-order method (or maximization method without derivatives). However, this simple search method may result in local optima. Thus, some advanced direct search methods can be applied in a Price Establishing Protocol to achieve global optimization with fast converging speeds, such as the Pattern Search Method (Kolda et al. 2003).

Next, we consider the computational complexity of this distributed algorithm. For each stage of this iterative algorithm, the pCP handles each cCP once, so each stage takes $O(m)$ computational steps where m is the number of cCPs. As we used a direct search method, at most $O(\epsilon^{-2})$ iterations or $O((b+1)\epsilon^{-2})$ objective function (Eq. 10) evaluations (when using positive spanning sets with b + 1 directions) are needed for a pCP to drive the norm of the gradient of the objective function below ϵ (for $\epsilon > 0$ arbitrarily small) (Vicente 2011).

Here, by $O(M)$, $M = \epsilon^{-2}$ means a multiple of M, where the constant multiplying of M does not depend on the dimension b of the problem or on the iteration counter k of the method under analysis (thus depending only on f or on algorithmic constants set at the initialization of the method). Thus, the overall computation time is $O(m + (b+1)\varepsilon^{-2})$.

4.2 Utility maximization game for cCPs (UtilMaxcCP)

In this resource allocation game, we assume that a pCP can set a fixed budget based on its price function so that the revenue it will give to cCPs is always lower than the revenue it can get from clients. Thus, in this game, the pCPs just announce their prices and do not intend to maximize their profit or utilities over cCPs. The cCPs in this game cooperatively try to determine their best coalition or Nash equilibrium point that maximizes their individual profits and thus achieves higher social welfare. Not all the cCPs participate in this game at a steady state. The formulation of this resource allocation game in a HDCF platform among seller cCPs can be described as follows:

1. **Players:** There are two types of players, the pCP and the available cCPs.
2. **Strategies:** The strategy for a pCP is to motivate the available CPs to form a HDCF platform to contribute resources by defining the price function $\mathrm{Pr}_{cCP}^t(R_{VM}^t)$. The strategy for the cCPs is to maximize the social welfare of the HDCF platform.
3. **Payoffs:** The payoff for the pCP is the utility it can gain by getting VM resources from available cCPs with the defined price function. For the cCPs, the payoff is the social welfare, which is defined as the total net utility summed over all cCPs i.e. $\sum_{i=1}^m Util(r_i^t)$

The global objective is to maximize the social welfare among the cCPs in the HDCF platform, that is,

$$Max \sum_{i=1}^m Util\left(r_i^t\right) = Max \sum_{i=1}^m \left[r_i^t \cdot \underset{cCP}{\overset{t}{\mathrm{Pr}}}\left(R_{VM}^t\right) - Cost\left(r_i^t\right)\right] \tag{31}$$

$$s.t.\quad 0 \leq r_i^t \leq \tilde{C}_i^t \tag{32}$$

We can simplify the objective function in Eq. 31 as follows:

$$Max \sum_{i=1}^{m} Util\left(r_i^t\right) = Max\left[R_{VM}^t \cdot \underset{cCP}{\overset{t}{\Pr}}\left(R_{VM}^t\right) - Min\ Cost\left(R_{VM}^t\right)\right] \quad (33)$$

$$s.\,t.\quad 0 \le r_i^t \le \tilde{C}_i^t,\ \ R_{VM}^t = \sum_{i=1}^{m} r_i^t,\ 0 \leqslant R_{VM}^t \leqslant \sum_{i=1}^{m} \tilde{C}_i^t \quad (34)$$

The first term $R_{VM}^t \cdot \underset{cCP}{\overset{t}{\Pr}}(R_{VM}^t)$ in Eq. 33 is the revenue the cCPs will get by providing R_{VM}^t resources, while the second term $\sum_{i=1}^{m} Cost(r_i^t)$ represents the total cost of providing R_{VM}^t units of VM by the cCPs. Now with the fixed R_{VM}^t quantity, the maximum social welfare can be obtained when the total cost of providing the VM resource is minimized. Once we can solve the minimum cost problem, we can also determine the optimal quantity of VM resource R_{VM}^t that results in maximum social welfare (Eq. 33). Now our goal is to find the minimum cost to supply a fixed quantity of VM resource R_{VM}^t among the cCPs, that is,

$$Min\ Cost\left(R_{VM}^t\right) \quad (35)$$

$$s.\,t.\quad 0 \le r_i^t \le \tilde{C}_i^t,\ \ R_{VM}^t = \sum_{i=1}^{m} r_i^t \quad (36)$$

As the function in Eq. 35 is a concave function. minimizing it over a linear convex region is complex. We can find a local minimum which is not necessarily a global minimum. To solve this problem algorithmically, we can use a dynamic programming method as described in Fontes et al. (2006). However, this is a centralized algorithm which requires the full knowledge of every cCPs cost functions to execute the algorithm. As a consequence, a distributive algorithm is generally preferred to solve the problem. A cCP coordinator is required to run the algorithm.

Similar to the approach mentioned earlier in Section 4.1, the cCP coordinator will take the role of a pCP and try to find out the minimum cost to obtain a fixed quantity of R_{VM}^t resource. The cCP coordinator introduces an initial shadow price σ and notifies the cCPs. Each cCP reacts to the shasow price with $r_i^t = \left[\frac{M_i^t}{\sigma}\right]^{-\frac{1}{\log_2 \alpha_i}}$ which is similar to the Eq. 20. The coordinator updates the shadow price based on the total VM resource contributed to the system. If the total VM resource supply is more than required, the σ value is reduced. If the VM supply is insufficient, the σ value should be increased. The process continues until the optimal value is achieved, i.e. $\sum_{i=1}^{m} r_i^t = R_{VM}^t$. The coordinator updates the shadow price as follows:

$$\sigma^{(n+1)} \leftarrow \sigma^{(n)} + \frac{\sigma^{(n)} \cdot \xi \cdot \left(R_{VM}^t - \sum_{i=1}^{m} r_i^t\right)}{R_{VM}^t} \quad (37)$$

The parameter n is the game stage and ξ is the factor that controls the converging speed of the algorithm. The method is executed periodically to ensure that the cost remains minimal after any cCPs join or leave. In a steady state, the σ leads to a VM allocation with minimal costs. The cCP coordinator has the additional task of determining whether the resource constraints of the cCPs are violated. If r_i^t is less than or equal to zero for a certain cCP i, the cCP is not eligible to contribute, as the cost of supplying the VM is comparatively high. Similarly, if r_i^t is greater than $\tilde{C}_i^t$, the cCP has an optimal value of r_i^t. This cCP should provide as much VM resource as possible since the cost is comparatively low. Thus, we can eliminate some cCPs, whose values of r_i^t are already known, and resolve the r_i^t for the remaining cCPs. The algorithm is described step by step as follows:

Step 1: Initialize $R_{VM}^{t'} = R_{VM}^t - \sum_{i=1}^{m} r_i^t$ as new VM resource requirement, the shadow price $\sigma = 0.1$, VM resource $r_i^t = 0$ and $P = \{P_i^t | i = 1...m\}$ is the set of available cCPs.

Step 2: Send $\sigma = 0.1$ to all cCPs and receive corresponding $r_i^t = \left[\frac{M_i^t}{\sigma}\right]^{-\frac{1}{\log_2 \alpha_i}}$

Step 3: If $R_{VM}^{t'} = \sum_{i=1}^{m} r_i^t$, then the algorithm finds the optimal r_i^t and break.

Otherwise, we have to adjust σ according to $\sigma \leftarrow \sigma + \frac{\sigma \cdot \xi \cdot \left(R_{VM}^{t'}\right)}{R_{VM}^t}$.

Step 4: If $0 \le r_i^t \le \tilde{C}_i^t$ for all $i \in P$, then break.

Step 5: Now for some cCP, $i \in P$, If $r_i^t \le 0$, remove those cCPs from the list of P. Also for some cCP, $i \in P$, If $r_i^t \ge \tilde{C}_i^t$, set $r_i^t = \tilde{C}_i^t$.

Now to determine the optimal quantity of VM resource R_{VM}^t that results in maximum social welfare in Eq. 33, we cannot rely on any optimization method that involves derivative of the objective function since we do not have the close form solution for the minimum cost problem, We use direct search method like pattern search method as described in Kolda et al. (2003) to find the optimal value of R^t_{VM} as follows:

Initialization

Let f denotes $\sum_{i=1}^{m} Util(r^t_i)$ or $Util(R^t_{VM})$ in Eq. 33. We have to maximize f in terms of R^t_{VM} s.t. $0 \leqslant R^t_{VM} \leqslant \sum_{i=1}^{m} \tilde{C}^t_i$.

Let Ω denotes the feasible region $(0 \leqslant R^t_{VM} \leqslant \sum_{i=1}^{m} \tilde{C}^t_i)$.

Let $R^{t^0}_{VM} \in \Omega$ be the initial guess.

Let $\Delta_{tol} > 0$ be the step-length convergence tolerance.

Let $\Delta_0 > \Delta_{tol}$ be the initial value of the step-length control parameter.

Let $\theta_{\max} < 1$ be an upper bound reduction parameter for the step-length control parameter.

Algorithm For each iteration k = 0, 1, 2, ...

Step 1: Let D_k be the set of search directions for iteration k. The search directions need to positively span the search space to generate the trial points.

Step 2: If there exists $d_k \in D_k$ such that trial point $f(R^{t^k}_{VM}+\Delta_k d_k) > f(R^{t^k}_{VM})$ or $f(R^{t^k}_{VM}-\Delta_k d_k) > f(R^{t^k}_{VM})$ and $(R^{t^k}_{VM}+\Delta_k d_k)$ or $(R^{t^k}_{VM}-\Delta_k d_k) \in \Omega$, then do the following:

- Set $R^{t^{k+1}}_{VM} = R^{t^k}_{VM} + \Delta_k d_k$ or $R^{t^k}_{VM} - \Delta_k d_k$ (change the iterate).
- Set $\Delta_{k+1} = \phi_k \Delta_k$, where $\phi_k \geq 1$ (optionally expand the step-length control parameter).

Step 3: Otherwise, either $(R^{t^k}_{VM} + \Delta_k d_k)$ or $(R^{t^k}_{VM} - \Delta_k d_k) \notin \Omega$ or $f(R^{t^k}_{VM} + \Delta_k d_k) \leq f(R^{t^k}_{VM})$ or $f(R^{t^k}_{VM} - \Delta_k d_k) \leq f(R^{t^k}_{VM})$, then do the following:

- Set $R^{t^{k+1}}_{VM} = R^{t^k}_{VM}$(no change the iterate).
- Set $\Delta_{k+1} = \theta_k \Delta_k$, where $0 < \theta_k < \theta_{max} < 1$ (reduce the step-length control parameter).
- If $R^{t^{k+1}}_{VM} < \Delta_{tol}$, then terminate.

The solution to the optimal VM quantity problem guarantes maximum social welfare among the cCPs in the HDCF platform. The optimal VM resource supplied by each cCP is determined through the minimum cost method and the price is set according to the price function.

Now let us consider the computational complexity of the UtilMaxcCP game. The objective of this game is to maximize the social welfare of the cCPs and two sub-problems must be solved: (1) For a fixed quantity of VM resource requirement, the minimum cost from cCPs that maximizes the social welfare must be determined. As each cCP must be handled once, each stage takes $O(m)$ computational steps. (2) The optimal quantity of the total VM resource that guarantees the best social welfare among the cCPs. As the direct search method like a pattern search method is used to solve the problem, at most $O(\epsilon^{-2})$ iterations or $O((b+1)\epsilon^{-2})$ objective function (Eq. 31) evaluations are needed for a cCP to drive the norm of the gradient of the objective function below ϵ (for $\epsilon > 0$ arbitrarily small). Therefore, the overall computation time is $O(m(b+1)\varepsilon^{-2})$.

4.2.1 Non-cooperative utility maximization game for cCPs (NonCopUtilMaxcCP)

We develop a non-cooperative utility maximization game for cCPs (Non-CopUtilMaxcCP) in an HDCF environment to show the effectiveness of *cooperation* among cCPs to achieve individual optimum as well as maximum social welfare in the cooperative UtilMaxcCP game.

In the NonCopUtilMaxcCP game of resource allocation, the cCPs make decision to maximize their own utilities regardless of other cCPs. They choose r^t_i based on the public information: the aggregated VM resource R^t_{VM} and the price function $\Pr^t_{cCP}(R^t_{VM})$ defined by a pCP and sequentially negotiate with each other to reach a Nash Equilibrium. There is no central authority in this game. Formally, cCP i needs to perform:

$$Max\ Util\left(r^t_i\right) = r^t_i \cdot \Pr^t_{cCP}\left(R^t_{VM}\right) - Cost\left(r^t_i\right) \tag{38}$$

$$s.t.\quad 0 \leq r^t_i \leq \tilde{C}^t_i \tag{39}$$

If the net utility of a cCP is less than or equal to zero, it will not participate in the game, and it will be removed from the list of cCPs. Note that R^t_{VM} is implicitly depends on r^t_i. If the value of r^t_i is changed, the value of R^t_{VM}, as well as $\Pr^t_{cCP}(R^t_{VM})$, will be changed accordingly. Thus, in the optimization, the value of R^t_{VM} would be better presented in terms of r^t_i. Let r^t_{-i} be the amount of VM resource collectively supplied by the cCPs except cCP i, then $r^t_{-i} = R^{t'}_{VM} - r^{t'}_i$, where $R^{t'}_{VM}$ and $r^{t'}_i$ are the total amount of VM resources and the amount of VM resources supplied by cCP i

respectively in the previous round. The equivalent optimization problem of Eq. 38 can be re-written by using Eqs. 2 and 6 as follows:

$$Max\ Util\left(r_i^t\right) = r_i^t \cdot M \cdot e^{-\left(r_i^t + r_{-i}^t\right)\cdot\omega} - \frac{M_i^t \cdot {r_i^t}^{1+\log_2\alpha_i}}{1+\log_2\alpha_i} \tag{40}$$

$$s.t.\quad 0 \le r_i^t \le \tilde{C}_i^t \tag{41}$$

Now to obtain the optimal value of (r_i^t) for each cCP i, we take the derivative of Eq. 40 with respect to r_i^t as follows:

$$Util'\left(r_i^t\right) = M \cdot e^{-\left(r_i^t+r_{-i}^t\right)\cdot\omega} - r_i^t \cdot M \cdot \omega \cdot e^{-\left(r_i^t+r_{-i}^t\right)\cdot\omega} - M_i^t \cdot {r_i^t}^{\log_2\alpha_i} = 0 \tag{42}$$

From Eq. 42, it is difficult to find the close form solution of optimal r_i^t for cCP i. A direct search method with linear constraint like pattern search method (Kolda et al. 2003), can be used with with multiple initial guesses for the optimal VM quantity r_i^t. In general, the game will converge to a Nash equilibrium.

However, for a defined price function, the Nash equilibrium of this non-cooperative game may not be *unique* as the order of movement will influence the equilibrium point. The game may converge to a different equilibrium, depending on the sequence of moves of the cCPs. So there is no guarantee that the equilibrium will be socially optimal. Hence, the outcome may not meet their expectation: that is, the utility may be worse than the achievable individual optimal, in which the cCPs cooperatively decide how many VM resources to supply.

For the non-cooperative situation, once the price function is set and publicized, the pCP has no control over the value of R_{VM}^t, which is the Nash equilibrium point resulting from the moves of the cCPs over many iterations.The non-cooperative game is not desirable for maximizing the pCP's profit because it does not lead to a unique Nash equilibrium. There is no guarantee for the pCPs to set a particular pricing function that leads to a desirable outcome, maximizing its total profit (see Eq. 10).

The computational complexity of the NonCopUtilMaxcCP game can be calculated as follows: In this game, the cCPs (m) sequentially negotiate with each other to reach a Nash Equilibrium based on the defined price function of the pCPs. There is no central authority. As the pattern search method is utilized by each cCP, the overall computation time is $O(m(b+1)\varepsilon^{-2})$.

5 Evaluation

In this section, we present our evaluation methodology and simulation results to show the effectiveness of the proposed cooperative resource allocation games- the utility maximization game for pCPs (UtilityMaxpCP) and the utility maximization game for cCPs (UtilityMaxcCP), in a HDCF platform. We used a mathematical simulation example to demonstrate how different CPs (pCPs and cCPs) interact with each other. We also analyzed a non-cooperative utility maximization game for cCPs (NonCopUtilMaxcCP) in this environment to show the necessity of cooperation. The cooperative games provide strong motivations to different CPs (pCPs and) to form or join in an HDCF platform. The mathematical simulation was implemented using MATLAB 7.0. The built-in Pattern Search Tool was used to solved the search problem.

5.1 Evaluation methodology

We considered an example scenario of one pCP and six cCPs in an HDCF platform and applied different resource allocation games to them. Since no real trace data of an HDCF platform have been published by any cloud provider, we conducted the experiments using synthetic data, as with other research (An et al. 2010; Antoniadis et al. 2010; Ardagna et al. 2011) . The experimental parameters used in the resource allocation games for cCPs are shown in Table 2. Using Amazon (EC2) as the example, we assumed that the range of production costs of first unit varied from 2$/hr to 3$/hr and that the service availability for all providers was 99.95% (Amit and Xia 2011). The learning factors were typically within the range (0.75, 0.9). For simplicity, we assumed that each cCP had almost the same amount of VM resource capacity in period t. Table 3 shows the parameters value used by a pCP in the resource allocation games to determine the price function. For all of the games, the converging speed factor ξ is set to 0.3.

Table 2 Parameters used in the resource allocation games for cCPs

cCPs i	Production cost of first unit per hr M_i^t	Learning factor α_i	Total capacity $\tilde{C}_i^t$
1	2.8	0.79	300
2	2.7	0.84	302
3	2.0	0.83	305
4	2.3	0.80	304
5	2.9	0.78	303
6	2.4	0.78	301

Table 3 Parameters used by a pCP in the resource allocation games

The rate of revenue function ω	Production cost of first unit per hr set for cCP M
0.01	3

The performance measures we considered for the games were: the social welfare, total profit, and return on investment, cost effectiveness and scalability. The social welfare defines the total value that an HDCF platform has for all participants, the return on investment is the ratio between the utility and the cost, and the cost effectiveness implies finding the best set of low-cost cCPs in the resource allocation games.

5.2 Simulation results

5.2.1 Convergence of the resource allocation games

Convergence is a basic requirement, as the VM resources allocated by the cCPs should converge in each game. We analyzed the behavior of each game based on the VM resources supplied at the steady state. The initial strategy of each cCP i is the zero vector. Each cCP then refines and updates its strategy at each iteration. We assumed that the cCPs updated their strategy in a sequential manner. Figures 3 and 4 depict the quantity of VM resources supplied by the six cCPs in each iteration of the three games.

As can be seen from the Figs. 3 and 4, the resource allocation games converged to a steady state after a number of iterations whereby no cCP has a tendency to unilaterally changes its strategy (a unique Nash equilibrium does exist in the game). However, changes to the strategy may be needed in the next period due to changes in the system's states (revenue function parameters, learning factor etc.). The algorithms are executed once by the players, which takes relatively little computation time.

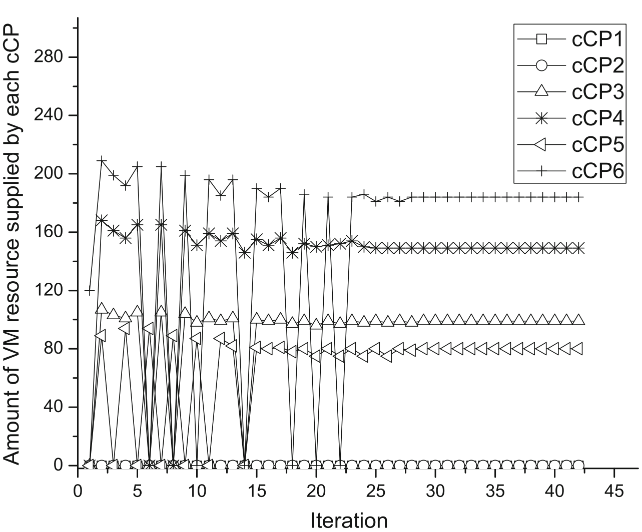


Fig. 3 VM resource supplied by each cCP in utility maximization game for pCPs

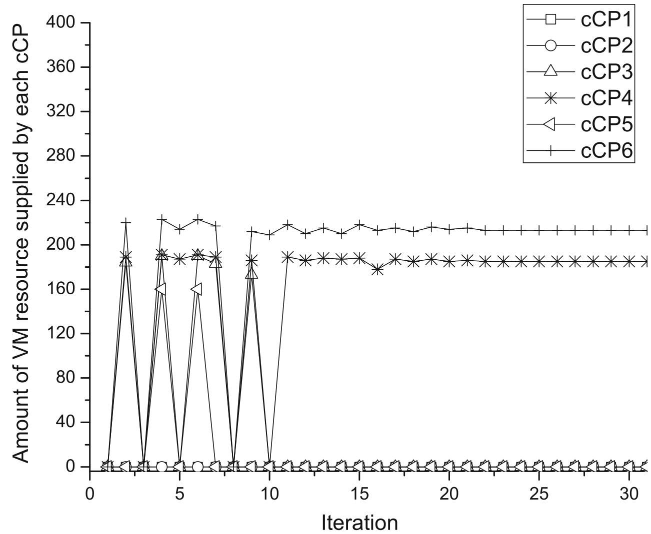


Fig. 4 VM resource supplied by each cCP in the utility maximization game for cCPs

5.2.2 Performance analysis of cooperative resource allocation games

We first evaluated the total profit in the cooperative resource allocation games. Figure 5 plots the total profit

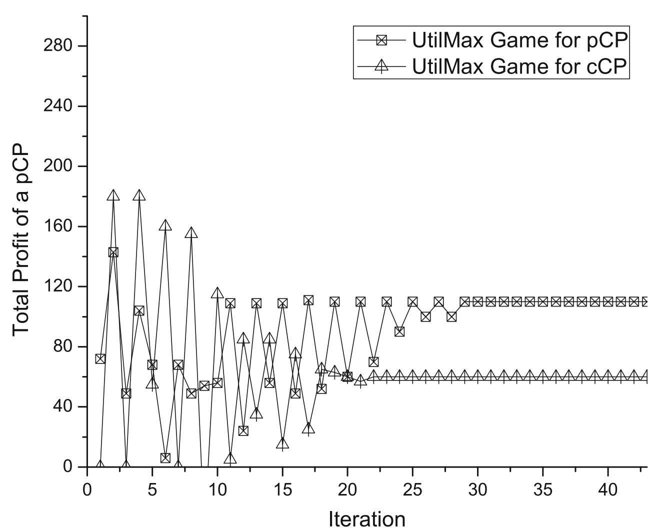


Fig. 5 Total profit in each resource allocation game

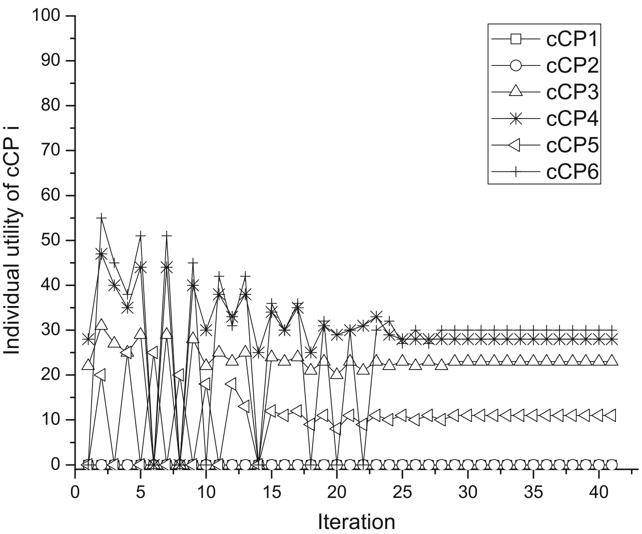


Fig. 6 Individual utility of cCPs in pCP utility maximization game

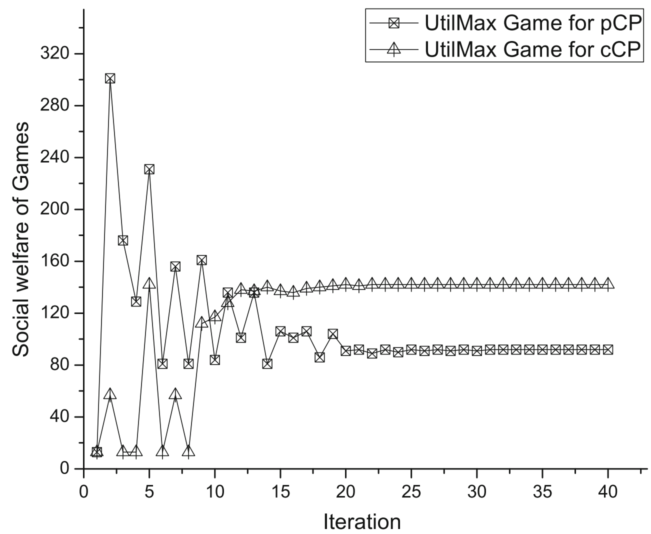


Fig. 8 Social welfare of cCPs in each resource allocation game

or the utility of the pCP in the two resource allocation games. The utility maximization game for the pCP generated a higher total profit (110) compared to the utility maximization game for the cCPs (60). The utility maximization game for the cCPs performed the worst, because it tried to maximize the social welfare in the platform by trading off the net profit of the pCP.

Next, we evaluate the social welfare and return on investment in the cooperative resource allocation games. Figures 6 and 7 demonstrate the individual utility of the cCPs. Only the cCPs with positive utility supplied VM resources to the HDCF platform. Figure 8 shows the social welfare in the two cooperative resource allocation games. The social welfare achieved by the UtilMaxcCP game is much higher (142) than the the UtilMaxpCP game (92). Furthermore, the maximum social welfare was achieved in the UtilMaxcCP game when the VM resource quantity was around 398 (see Fig. 4). The return on investment in the UtilMaxcCP game was 6.2. In contrast, the return on investment for the UtilMaxpCP game was 2.4. The UtilMaxcCP game was more effective in terms of cost, as few low-cost cCPs (cCP6 and cCP4) provided more VM resources

We subsequently investigated the scalability issue for both resource allocation games. We varied the number of cCPs available in the HDCF system from 6 to 24 for ω value 0.01 and evaluated its effect on the convergence

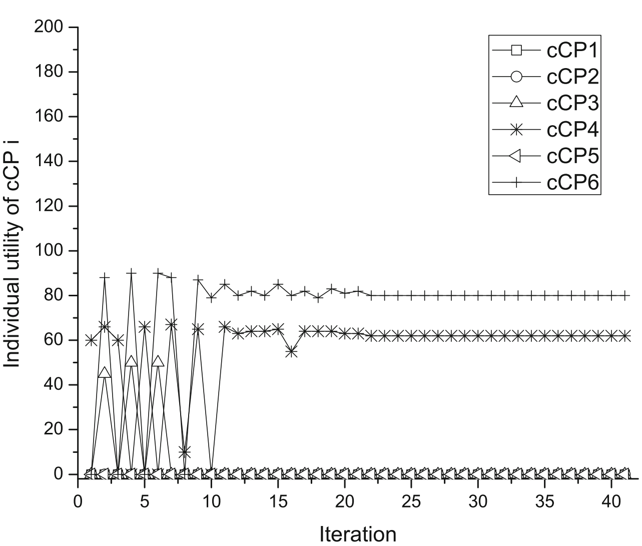


Fig. 7 Individual utility of cCPs in cCP utility maximization game

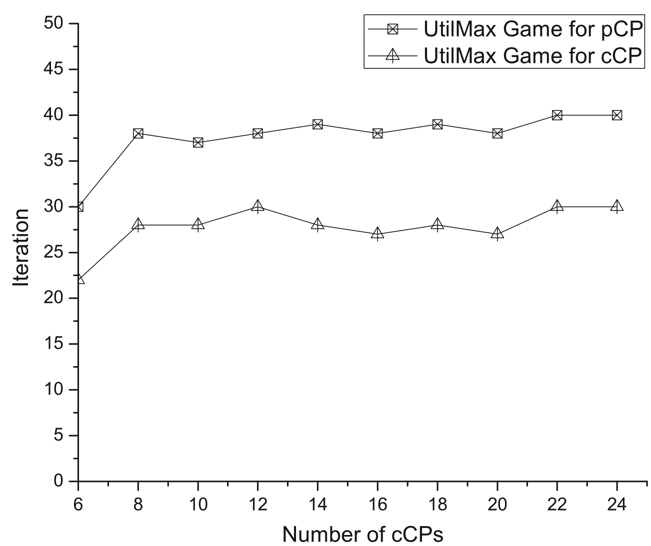


Fig. 9 Performance of two resource allocation games in terms of convergence with different number of cCPs

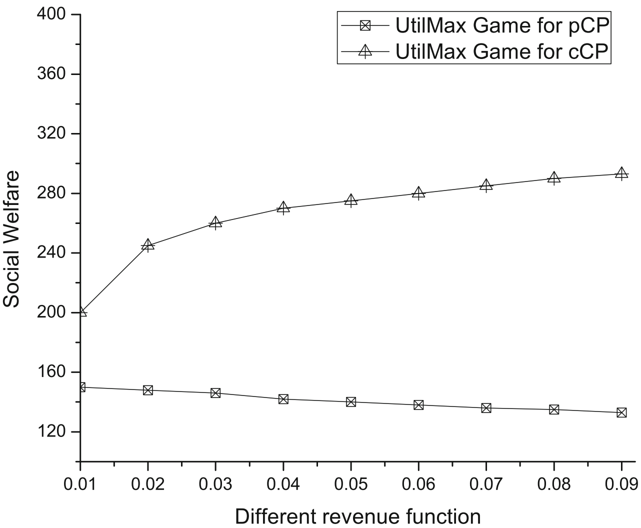


Fig. 10 Performance of two resource allocation games in terms of social welfare with 24 cCPs under different revenue function

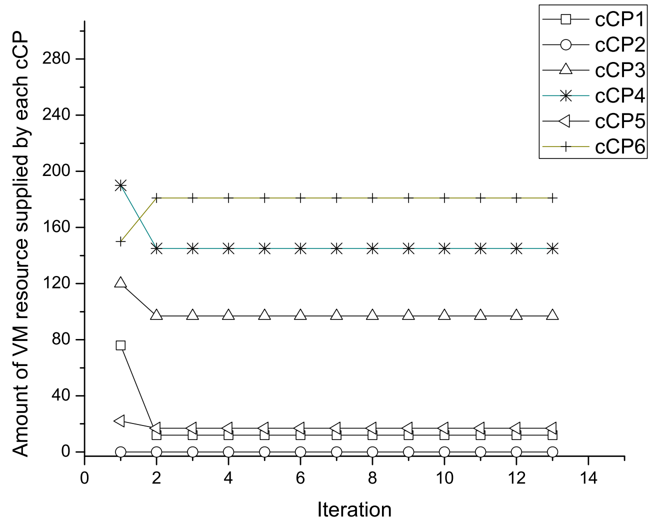


Fig. 11 VM resource supplied by each cCP in NonCopUtilMax-cCP game

of the two games. The six types of cCPs (shown in the Table 2) were preserved in this simulation. Figure 9 shows the number of iterations required to reach equilibrium as the number of cCPs in the HDCF system increased in both the games. The number of iterations required remained relatively stable for both games as the number of cCPs varied between 6 and 24. However, the utility maximization game for cCPs converged fast, as only the cCPs that provided cost-effective VMs remained in a steady state.

We also evaluated the scalability of the two games in terms of social welfare with different revenue function ω. Here we considered the number of cCPs to be 24. The larger the ω the higher the revenue was for the same quantity of VM resource. A small ω implies that more VM resource is required to obtain the same amount of revenue (Fig. 10). The UtilMaxcCP game outperformed the other game in social welfare maximization. Moreover, the social welfare of the UtilMaxpCP game decreased with as ω. increased.

5.3 Discussion

From the simulation results, we can see that proposed cooperative game settings motivate different self-interested CPs (pCPs and cCPs) to form or join a HDCF platform. In fact, the UtilMaxpCP game maximized total profit for pCPs whereas the UtilMaxcCP game maximized sofial welfare for cCPs. In general, the UtilMaxpCP game, which is a Stackelberg game, is suitable for a system that contains a centralized authority, like the pCP, and the UtilMaxcCP game is good for a non-coordinated P2P-like application. As a HDCF platform becomes decentralized (i.e. there is no fixed central authority or leader) and the pCP or cCPs join or leave dynamically, the UtilMaxpCP game is more suitable as a distributed resource allocation game. Since this game achieved higher social welfare and return on investment, it strongly motivated the cCPs to participate in an HDCF platform. Also, this game provided effective resource supply in terms of cost to an HDCF platform and thus admits the best set of cCPs to participate.

We subsequently compared the performance of the UtilMaxcCP game with the non-cooperative utility maximization game for cCPs (NonCopUtilMaxcCP) in

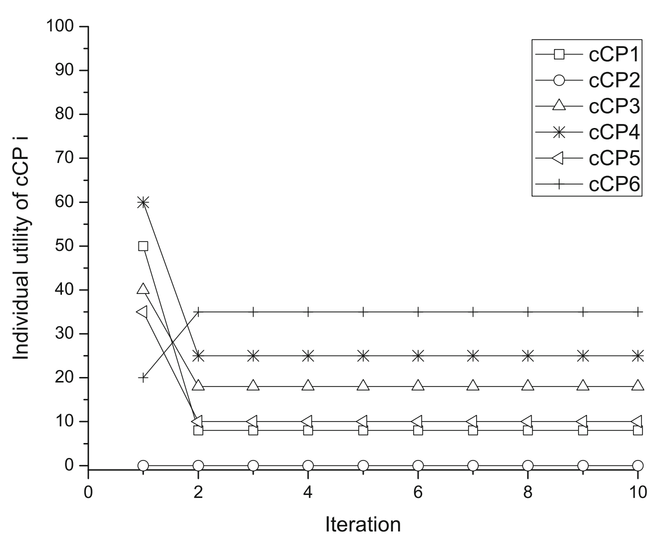


Fig. 12 Individual utility of cCPs in NonCopUtilMaxcCP game

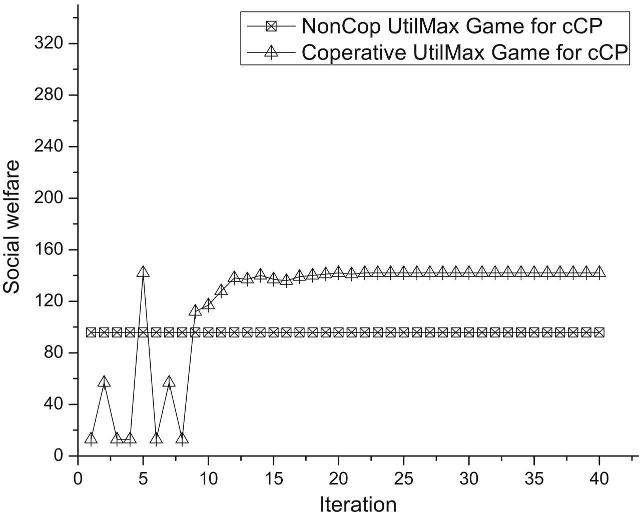


Fig. 13 Social welfare of cCPs in the NonCopUtilMaxcCP game and cooperative UtilMaxcCP game

a HDCF environment to show the effectiveness of *co-operation* among cCPs to achieve individual optimum as well as maximum social welfare. The same parameter settings were used (shown in Table 2). Figures 11, 12 and 13 show the experimental results. The NonCopUtilMaxcCP game converged based on the VM resources supplied at the steady state (Fig. 11).

From Figs. 7 and 12 we can see that the individual utility of cCPs in the NonCopUtilMaxcCP game was much lower than that of in the cooperative UtilMaxcCP game. The social welfare was also at a maximum in the cooperative UtilMaxcCP game (Fig. 13). These results demonstrate that in a non-cooperative environment, the cCPs selfishly optimize their own utilities. As a result, the optimal aggregated benefits received by the cCPs are not guaranteed. The outcome may not meet their expectations and the utility may be worse than the achievable individual optimal, in which the cCPs would cooperatively decide how many VM resources to supply.

5.4 Practical and theoretical implications of the proposed game theoretic approaches

Modern clouds function in an open world characterized by continuous changes which occur autonomously and unpredictably. They are rational (i.e. self-interested and welfare-maximizing) and make individual decisions based on local knowledge and preferences without considering the global good. The global efficiency is generated through interactions among CPs. In this context, the proposed cooperative game theoretical methods allow for an in-depth analytical understanding of the distributed VM resource provisioning problem in an HDCF platform, as they allow us to account for the inherently contradictory interests of the resource users (pCPs) and the resource providers (cCPs).

In addition, our proposed cooperative game theoretical approaches of resource allocation can help a CP to determine rational strategies for pricing and resource provision decisions. They enable a quantitative framework for a CP for obtaining management solutions and encourage a CP to learn and react to the critical parameters in the operation management process by gaining useful business insights. Furthermore, the use of a learning curve model in the cost functions of a CP can help to understand customers' cloud adoption decisions and explain quantitatively why cloud computing is the most attractive to small and medium businesses.

6 Conclusions and future works

In this paper, we analyzed cooperative game theory based optimal VM resource allocation mechanisms among IaaS CPs (pCPs and cCPs) with a heterogeneous cost function in a HDCF environment. In particular, we studied the question of motivation for each self-interested IaaS CP, motivating them to form or join an HDCF platform and how much VM resource should be allocated. We proposed two utility maximizing cooperative games, one for pCPs and the other for cCPs, that led to two different optimal situations: maximized total profit for pCPs and maximized social welfare for cCPs. By evaluating the total profit and the social welfare received by the CPs (pCPs and cCPs), we demonstrated that the proposed game settings motivated different CPs to cooperate in an HDCF platform. Both centralized and distributed algorithms with guaranteed convergence were presented to determine optimal solutions. We also carried out extensive simulations to measure the effectiveness of these algorithms in an HDCF platform. Under a utility maximization game for pCPs, desirable outcomes (e.g. social welfare, return on investment etc.) cannot be maximized. However, under the utility maximization game for the cCPs, the IaaS CPs were strongly motivated to contribute VM resources among themselves. Also, this game was effective in terms of cost and was scalable, as only the collaborators with low-costs participated in an HDCF platform. We also compared its performance with a non-cooperative utility maximization game for cCPs and showed that no system-wide property (e.x. individual optimum, maximum social welfare etc.) was achieved.

Our paper opens up several exciting avenues for future works, such as:

- evaluating the performances of resource allocation games in real HDCF platforms where hundreds of clouds dynamically join and leave the federation
- improving utility functions of CPs for more efficient negotiations and extending their terms to include other economic or performance goals
- finding methods for the maximization of utility functions that include fuzzy values for non-deterministic data
- considering the dependencies across multiple types of applications hosted in a HDCF environment
- developing local task and resource allocation mechanisms that interact with HDCF platforms from CPs
- developing a single sign on authentication mechanism for enabling a HDCF platform, establishing trust contexts between different CPs
- developing information and data models that can capture the concepts and semantics of the resources and services offered by HDCF platforms

Acknowledgement This work was supported by a post-doctoral fellow ship grant from the Kyung Hee University, Global Campus, Korea in 2011 (KHU-20110217).

References

Amit, G., & Xia, C. H. (2011). Learning curves and stochastic models for pricing and provisioning cloud computing services. *Service Science, 3*, 99–109.

An, B., Lesser, V., Irwin, D., & Zink, M. (2010). Automated negotiation with decommitment for dynamic resource allocation in cloud computing. In *Proceedings of the 9th international conference on autonomous agents and multiagent systems, AAMAS '10* (Vol. 1, pp. 981–988).

Andrade, N., Brasileiro, F., Cirne, W., & Mowbray, M. (2007). Automatic grid assembly by promoting collaboration in peer-to-peer grids. *Journal of Parallel and Distributed Computing, 67*, 957–966.

Antoniadis, P., Fdida, S., Friedman, T., & Misra, V. (2010). Federation of virtualized infrastructures: Sharing the value of diversity. In *Proceedings of the 6th international conference, Co-NEXT '10* (pp. 12:1–12:12). New York: ACM.

Ardagna, D., Panicucci, B., & Passacantando, M. (2011). A game theoretic formulation of the service provisioning problem in cloud systems. In *Proceedings of the 20th international conference on World Wide Web, WWW '11* (pp. 177–186).

Assunção, M. D., Costanzo, A., & Buyya, R. (2010). A cost-benefit analysis of using cloud computing to extend the capacity of clusters. *Cluster Computing, 13*, 335–347.

Auyoung, A., Chun, B., Snoeren, A., & Vahdat, A. (2004). Resource allocation in federated distributed computing infrastructures. In *Proceedings of the 1st workshop on operating system and architectural support for the on-demand IT infrastructure*. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.2.2369.

Bittman, T. (2008). *The evolution of the cloud computing market*. Gartner Blog Network, http://blogs.gartner.com/thomasbittman/2008/11/03/theevolution-of-the-cloud-computing-market/.

Buyya, R., Ranjan, R., & Calheiros, R. (2010). Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services. In *Algorithms and architectures for parallel processing. Lecture notes in computer science* (Vol. 6081, pp. 13–31).

Carroll, T. E., & Grosu, D. (2010). Formation of virtual organizations in grids: A game-theoretic approach. *Concurrency and Computation: Practice and Experience, 22*, 1972–1989.

Celesti, A., Tusa, F., Villari, M., & Puliafito, A. (2010a). How to enhance cloud architectures to enable cross-federation. In *IEEE international conference on cloud computing* (pp. 337–345).

Celesti, A., Tusa, F., Villari, M., & Puliafito, A. (2010b). Three-phase cross-cloud federation model: The cloud sso authentication. In *International conference on advances in future internet* (pp. 94–101).

Cheng, W. K., Ooi, B. Y., & Chan, H. Y. (2010). Resource federation in grid using automated intelligent agent negotiation. *Future Generations Computer Systems, 26*, 1116–1126.

Costanzo, A. d., Jin, C., Varela, C. A., & Buyya, R. (2009). Enabling computational steering with an asynchronous-iterative computation framework. In *Proceedings of the 2009 fifth ieee international conference on e-Science, E-SCIENCE '09* (pp. 255–262). Washington, D.C.: IEEE Computer Society.

di Costanzo, A., de Assuncao, M. D., & Buyya, R. (2009). Harnessing cloud technologies for a virtualized distributed computing infrastructure. *IEEE Internet Computing, 13*, 24–33.

Dodda, R. T., Smith, C., & Moorsel, A. (2009). An architecture for cross-cloud system management. In *Contemporary computing. Communications in computer and information science* (Vol. 40, pp. 556–567). Berlin: Springer.

Drew, F., & Jean, T. (1993). *Game theory*. Cambridge, MA: The MIT Press, ISBN-10: 0-262-06141-4

Elmroth, E., & Larsson, L. (2009). Interfaces for placement, migration, and monitoring of virtual machines in federated clouds. In *GCC '09: Proceedings of the 2009 eighth international conference on grid and cooperative computing* (pp. 253–260).

Feldman, M., Lai, K., Stoica, I., & Chuang, J. (2004). Robust incentive techniques for peer-to-peer networks. In *Proceedings of the 5th ACM conference on electronic commerce, EC '04* (pp. 102–111)

Fontes, D. B. M. M., Hadjiconstantinou, E., & Christofides, N. (2006). A dynamic programming approach for solving single-source uncapacitated concave minimum cost network flow problems. *European Journal of Operational Research, 174*, 1205–1219.

Fu, Y., Chase, J., Chun, B., Schwab, S., & Vahdat, A. (2003). Sharp: An architecture for secure resource peering. In *Proceedings of the nineteenth ACM symposium on Operating systems principles* (pp. 133–148). New York: ACM.

Goiri, I., Guitart, J., & Torres, J. (2010). Characterizing cloud federation for enhancing providers' profit. In *IEEE international conference on cloud computing* (pp. 123–130).

Gomes, E. R., Vo, Q. B., & Kowalczyk, R. (2010). Pure exchange markets for resource sharing in federated clouds. *Concurrency and Computation: Practice and Experience*. doi:10.1002/cpe.1659.

He, L., & Ioerger, T. R. (2005). Forming resource-sharing coalitions: A distributed resource allocation mechanism for

self-interested agents in computational grids. In *Proceedings of the 2005 ACM symposium on applied computing, SAC '05* (pp. 84–91).
Jalaparti, V., Nguyen, G. D., Gupta, I., & Caesar, M. (2010). *Cloud resource allocation games*. Technical Report, University of Illinois. http://hdl.handle.net/2142/17427.
Khan, S. U., & Ahmad, I. (2006). Non-cooperative, semi-cooperative, and cooperative games-based grid resource allocation. In *Proceedings of the 20th international conference on parallel and distributed processing, IPDPS'06* (pp. 121–121).
Kolda, T. G., Lewis, R. M., & Torczon, V. (2003). Optimization by direct search: New perspectives on some classical and modern methods. *SIAM Review, 45*, 385–482.
Kumar, C., Altinkemer, K., & De, P. (2011). A mechanism for pricing and resource allocation in peer-to-peer networks. *Electronic Commerce Research and Applications, 10*, 26–37.
Lai, K., Rasmusson, L., Adar, E., Zhang, L., & Huberman, B. A. (2005). Tycoon: An implementation of a distributed, market-based resource allocation system. *Multiagent and Grid Systems, 1*, 169–182.
Lee, C., Suzuki, J., Vasilakos, A., Yamamoto, Y., & Oba, K. (2010). An evolutionary game theoretic approach to adaptive and stable application deployment in clouds. In *Proceeding of the 2nd workshop on bio-inspired algorithms for distributed systems* (pp. 29–38). New York: ACM.
Li, M., Chen, M., & Xie, J. (2010). Cloud computing: A synthesis models for resource service management. In *2010 second international conference on communication systems, networks and applications (ICCSNA)* (Vol. 2, pp. 208–211).
Ma, R., Lee, S., Lui, J., & Yau, D. (2006). Incentive and service differentiation in P2P networks: A game theoretic approach. *IEEE/ACM Transactions on Networking, 14*(5), 978–991.
Macias, M., & Guitart, J. (2010). Using resource-level information into nonadditive negotiation models for cloud market environments. In *2010 IEEE network operations and management symposium (NOMS)* (pp. 325–332).
Maximilien, E. M., Ranabahu, A., Engehausen, R., & Anderson, L. (2009). Ibm altocumulus: A cross-cloud middleware and platform. In *OOPSLA '09: Proceeding of the 24th ACM SIGPLAN conference companion on object oriented programming systems languages and applications* (pp. 805–806).
OpenQRM (2010). *The next generation, open-source data-center management platform*. http://www.openqrm.com/.
Park, J., & van der Schaar, M. (2010). A game theoretic analysis of incentives in content production and sharing over peer-to-peer networks. *IEEE Journal of Selected Topics in Signal Processing, 4*(4), 704–717.
Penmatsa, S., & Chronopoulos, A. T. (2011). Game-theoretic static load balancing for distributed systems. *Journal of Parallel and Distributed Computing, 71*, 537–555.
Ranjan, R., & Buyya, R. (2008). Decentralized overlay for federation of enterprise clouds. CoRR abs/0811.2563.
Rochwerger, B., & Breitgand, D. (2009). The reservoir model and architecture for open federated cloud computing. *IBM Journal of Research and Development, 53*(4), 535–545.
Subrata, R., & Zomaya, A. Y. (2008). Game-theoretic approach for load balancing in computational grids. *IEEE Transactions on Parallel and Distributed Systems, 19*, 66–76.
Teng, F., & Magoulès, F. (2010). A new game theoretical resource allocation algorithm for cloud computing. In *Advances in grid and pervasive computing. Lecture notes in computer science* (Vol. 6104, pp. 321–330). Berlin: Springer.
Vicente, L. N. (2011). *Worst case complexity of direct search*. Department of Mathematics, University of Coimbra. http://www.mat.uc.pt/~lnv/papers/complex.pdf.
Wei, G., Vasilakos, V., Zheng, Y., & Xiong, N. (2010). A game-theoretic method of fair resource allocation for cloud computing services. *Journal of Supercomputing, 54*, 252–269.
Wilkins, J., Ahmad, I., Fahad Sheikh, H., Faheem Khan, S., & Rajput, S. (2010). Optimizing performance and energy in computational grids using non-cooperative game theory. In *Proceedings of the international conference on green computing, GREENCOMP '10* (pp. 343–355).
Wolski, R., Brevik, J., Plank, J. S., & Bryan, T. (2003). Grid resource allocation and control using computational economies. In *Grid computing: making the global infrastructure a reality* (pp. 747–772). New York: Wiley.

Mohammad Mehedi Hassan is an Assistant Professor of Information System Department and Chair of Pervasive and Mobile Computing, CCIS, at the King Saud University, Riyadh, KSA. He received his Ph.D. degree in Computer Engineering from Kyung Hee University, South Korea in 2010. He was a Research Professor at Computer Engineering department, Kyung Hee University, South Korea from March, 2011 to October, 2011. His research interests include Cloud collaboration, media Cloud, sensor-Cloud, mobile Cloud, Thin-Client, Grid computing, IPTV, virtual network, sensor network, and publish/subscribe system. He has authored and co-authored more than 43 publications including refereed IEEE/ACM/Springer journals, conference papers, books, and book chapters.

M. Shamim Hossain is an A/Professor of CCIS, at the King Saud University, Riyadh, KSA. He received his Ph.D. in Electrical and Computer Engineering from the University of Ottawa, Canada. His research interests include Service oriented computing, Collaborative Cloud media, Service configuration, and biologically inspired approach for multimedia and distributed system. He served as guest editor for IEEE Transactions on Information Technology in Biomedicine, and Springer Multimedia tools and Applications. He is a Senior Member of IEEE and a member of ACM.

A. M. Jehad Sarkar received Ph.D. in Computer Engineering from Kyung Hee University, Korea in 2010. Currently, he has been an Assistant Professor at the Dept. of Digital Information Engineering, Hunkuk University of Foreign Studies, Korea. His research interests are Activity Recognition, Web mining, Data mining and Cloud computing.

Eui-Nam Huh has earned BS degree from Busan National University in Korea, Master's degree in Computer Science from University of Texas, USA in 1995 and Ph. D degree from the Ohio University, USA in 2002. He was a director of Computer Information Center and Assistant Professor in Sahmyook University, South Korea during the academic year 2001 and 2002. He has also served for the WPDRTS/IPDPS community as program chair in 2003. He is the chief editor of Journal of Korean Society for Internet Information, editor of TIIS journal, and Grid Standard Project Group chair, TTA, Korea. He was also an Assistant Professor in Seoul Women's University, South Korea. Now he is with Kyung Hee University, South Korea as Professor in Dept. of Computer Engineering. His interesting research areas are: Cloud Computing, High Performance Network, Sensor Network, Distributed Real Time System, Grid Middleware, and Network Security.