Incentive Mechanisms for Opportunistic Cloud Computing Services

Eric Kuada, Henning Olesen
Center for Communication, Media and Information Technologies
Aalborg University
Copenhagen, Denmark
{kuada, olesen}@cmi.aau.dk

***Abstract*— Opportunistic Cloud Computing Service (OCCS) is a social network approach to the provisioning and management of cloud computing services for enterprises. The OCCS network may suffer from the free riding problem where members are selfish and will only want to use services on the platform without ever contributing resources. It may also suffer from resource wastage from members or external entities trying to attack the system so that genuine users are deprived of valuable resources. The purpose of this paper is to design incentive schemes that will encourage the contribution of resources to the OCCS platform as well as the efficient usage of these resources. We employ game theory and mechanism design to model and design the incentive schemes. We present two game models and show the existence of a pure strategy Nash equilibrium for both the cooperative and non-cooperative games. Three base incentive schemes are presented and two advanced schemes one based on discount factor and the other a stochastic scheme are also presented. We perform analytical evaluation of our incentive schemes and conclude that the schemes meet the desired properties of budget-balance, ex-post individual rationality, incentive compatibility, allocative efficiency, robustness, and flexible to accommodate changing user behavior on the platform.**

Keywords- game theory; mechanism design; opportunistic cloud computing services

I. INTRODUCTION

A lot of Small and Medium size Enterprises (SMEs) and even relatively big companies lack the adequate Information Technology (IT) resources necessary for their business processes. Spare IT resources such as computing power (CPU and RAM) and storage capacity are however available at some large corporations and even relatively smaller companies. It should be useful to have a way by which companies and other organizations can make use of these spare resources to get their required IT resources necessary for their business processes. The advent of cloud computing within the past few years technically supports this concept. Cloud computing is mainly the packaging of traditional information technology infrastructure and software solutions such as storage, CPU, network, applications , services, etc. as virtualized resources and delivered by a service provider to its customers as an on-demand pay-per-use self-provisioned service through a web portal over a network such as the Internet. There have been major technological advancements as well as social and business demands driving this new trend of computing. The technological factors facilitating cloud computing include the availability and drastic increase in reliable broadband Internet access, advancements in virtualization technologies and the shift of development of majority of both desktop and enterprise applications as web services and applications software.

Our hypothesis is that a social network approach can be used for companies with underutilized spare resources to provide IT resources to SMEs and other organizations that may need them by leveraging the cloud computing paradigm and supporting technologies. Though the concept of taking advantage of spare computing resources to meet certain demands – generally referred to as volunteer computing (e.g. Folding@home, SETI@home, and MilkyWay@home) has existed for a some years now, it focuses on gathering large numbers of donated computing cycles to form a large-scale virtual supercomputer; and only has support for embarrassingly parallel applications and does not offer enough varied and enticing alternative use-cases [1].

The goal therefore is to identify the sufficient enabling conditions for the successful implementation of this social network approach to IT resource sharing. Reference [2] introduced the Opportunistic Cloud Computing Services (OCCS) concept, presented the research issues that are needed to be addressed for its successful implementation and presented the detailed reference architect for it. Because OCCS promises an accelerated adoption of cloud computing services and further reduction in IT cost for small companies, we have been working on the feasibility of its successful implementation in terms of the technical feasibility, impact of public policy and regulations on its implementation, and its acceptance and support by all stakeholders. Additionally the data center and Cloud management competences that companies develop over time through managing their own resources can be of value to others lacking such competences (e.g. SMEs needs, especially in the developing world). OCCS also has the potential of fostering business collaboration, offering further reduction of cost in IT services and by design is compatible with future cloud computing technologies and solutions.

The OCCS network like peer-to-peer networks may suffer from the free riding problem where members are selfish and will only want to use services on the platform without ever contributing resources to it. It may also suffer from resource wastage from members on the platform or external entities trying to attack the system so that genuine

COLLABORATECOM 2012, October 14-17, Pittsburgh, United States

DOI 10.4108/icst.collaboratecom.2012.250403

users are deprived of valuable resources. The purpose of this paper is therefore to design incentive schemes that will encourage the contribution of resources to the OCCS platform as well as the efficient usage of these resources.

Additionally there should be enough motivation for managing the OCCS governance platform. Furthermore because enterprises will be making use of these resources on the platform to offer services to their customers, they may require some level of quality of service (e.g. availability). However, unlike as is the case with services provided by commercial cloud service providers, no SLA exists among the members on the OCCS network. It is therefore necessary to devise a way for offering services differentiation so that when scarce resources are being contended for by multiple members those that have contributed more resources to the OCCS network can be given some level of preferential treatment in offering these scarce resources to them.

The approach to addressing the challenges that are of interest in this work is to employ game theory [3] [4] to model and design incentive mechanisms that will encourage the contribution of resources to the OCCS platform as well as their efficient usage. The contribution of this work is the design of suitable incentive schemes for resource sharing of generic cloud computing services, and also providing the foundation for the implementation of Pseudo Service Level Agreement (Pseudo SLA) on the OCCS platform which is discussed later in the sequel to this paper on “Trust engineering and Pseudo SLA in Opportunistic Cloud Computing Services”. It should be evident from the title of the sequel to this paper that we consider data privacy, security and trust as major issues that OCCS and cloud computing in general has to address. We are therefore currently working on trust engineering in cloud computing and how to adapt it for the OCCS environment.

The remainder of the paper is organized as follows: Section II gives the background for this work, the first part of which is an overview on opportunistic cloud computing services and the second part discusses related works. Section III deals with the incentives modeling process and the results obtained. We present two game models and show the existence of a pure strategy Nash equilibrium for both the cooperative and non-cooperative games. Section III also presents a base incentive scheme with its three variations and two advanced schemes one based on discount factor and the other a stochastic scheme. Section IV presents the resource allocation process and the evaluation of the presented incentive mechanisms. Section V concludes the paper and also touches on future work. We also include an appendix on the proof of the existence of a pure strategy Nash equilibrium.

II. Background

The first part of this section gives an overview on cloud computing mainly on the types of services and the deployment models. The second part gives an overview of the OCCS concept regarding its main components. Its details and reference architecture can be found in [2]. The possible impact that public policy and data privacy regulations may have on its implementation can also be found in [5]. In the final part of this section we consider some related works that has a bearing on a at least two of the topical areas of cloud computing, resource allocation, game theory, and incentive mechanism design.

A. Cloud Computing Overview

Cloud computing is essentially the packaging of traditional information technology infrastructure and software solutions such as storage, CPU, network, applications, services, etc. as virtualized resources and delivered by a service provider to its customers as an on-demand pay-per-use self-provisioned service through a web portal over a network such as the Internet. There have been major technological advancements as well as social and business demands driving this new trend of computing. The technological factors facilitating cloud computing include the availability and drastic increase in reliable broadband Internet access, advancements in virtualization technologies and the shift of development of majority of both desktop and enterprise applications as web services and applications.

The three main components of a regular computing environment, namely the hardware infrastructure, the operating system platform and user application software, have respectively translated into Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) delivered in cloud computing. Additionally, there is an inexhaustible list of other cloud computing services due to the concept of “Anything as a Service” (XaaS) being the main driving idea of cloud computing. Thus, virtually all IT products and solutions are potential cloud computing services. These services are normally deployed in four main cloud deployment models namely public, private, community, and hybrid cloud computing deployment models [6].

A public cloud is one in which the infrastructure and other computational resources that it comprises are made available to the general public over the Internet. It is owned by a cloud service provider selling cloud services and by definition, is external to an organization. Private clouds are at the other end of the spectrum; a private cloud is one in which the computing environment is operated exclusively for an organization. It may be managed either by the organization itself or a third party such as a commercial cloud services provider, and may be hosted within the organization’s data center or outside of it. The community clouds and hybrid clouds fall between public and private cloud deployment models. A community cloud is similar to a private cloud, but the infrastructure and computational resources are shared by several organizations that have common privacy, security, and regulatory considerations, rather than for the exclusive use of a single organization. A hybrid cloud deployment model is a combination of two or more of the other cloud deployment models (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technologies that enable interoperability.

B. Opportunistic Cloud Computing Services

Opportunistic cloud computing service is a social network approach to the provisioning and management of cloud computing services for enterprises. OCCS deals with the concept of enterprises taking advantage of cloud computing services to meet their business needs without having to pay or paying a minimal fee for these services [2]. The OCCS network is a social network of enterprises collaborating strategically for the contribution and usage of cloud computing services without entering into any business agreements. Unlike social networking services provide by social networking sites for individual use where users create their own network of friends, in an OCCS network, members do not explicitly create ties with other members but these ties come indirectly through the resource contribution and consumption process.

The OCCS network platform is a governing platform that serves as the social networking platform for enterprises and also includes interoperable Cloud management tools with which member enterprises can provision resources that will be used by other enterprises interested in these services. The OCCS platform consists of two main layers – the service layer and the management layer. The service layer consists of all the services contributed by members. These will normally be fundamental cloud computing services such as SaaS, PaaS, and IaaS; but it can also include value added services normally provided by cloud service brokers. The management layer consists of two main components – the governance component that manages the services from members and cloud services brokerage component that serves as an interface between the OCCS network and commercial cloud services providers and cloud service brokers.

C. Related Work

It this section we consider related works that has a bearing on at least two of the topical areas of cloud computing, resource allocation, game theory, and incentive mechanism design. In [7] they looked at a single commercial IaaS provider service provisioning problem for multiple SaaS providers who in turn are trying to maximize their profit on services provided to their end-user customers subject to SLA constraints. SaaS providers want to maximize their revenues from SLAs, while minimizing the cost of use of resources supplied by the IaaS provider. They modeled the service provisioning problem as a Generalized Nash game, and proposed an algorithm for the run time management and allocation of IaaS resources to competing SaaS providers.

Reference [8] proposed the incorporation of mechanism design to enforce and achieve efficient resource utilization among selfish VMs in non-cooperative cloud systems. They looked at the problem that on cloud platforms, computing resources are allocated on-demand dynamically and the application hosted on a virtual machine (VM) usually has the illusion of complete control of resources. Thus, a selfish VM may strategically compete for resource with other VMs to maximize its own benefit while at the cost of overall system performance; this problem poses new challenges to cloud providers, who must thwart non-cooperative behavior as well as allocating resource among selfish VMs efficiently. They proposed to use mechanism design to allocate resource among selfish VMs in a non-cooperative cloud environment.

Reference [9] considered applying game theory to the resource allocation problem where service demanders intend to solve parallel computing problem by requesting the usage of resources across a cloud-based network while service providers schedule and allocate resources to the individual subtasks. Other related works such as [10] have used game theory in the modeling of incentives in P2P systems.

The related works mentioned above and other previous works on resource allocation and mechanism design on cloud services all look at the resource allocation problem of already existing resources of commercial cloud services. We are in addition to this, looking at incentive mechanism design for the schemes to also promote the contribution of resources to the platform together with the efficient usage of these contributed resources.

III. INCENTIVE MECHANISMS FOR OCCS

Mechanism design is the sub field of microeconomics and game theory that considers how to implement good system-wide solutions to problems that involve multiple self-interested agents, each with private information about their preferences. It is best to view the goals of the designed mechanisms in the very abstract terms of social choice. A social choice is an aggregation of the preferences of the different participants towards a single joint decision. Mechanism design attempts to implement desired social choices in a strategic setting. Such strategic design is necessary since usually the preferences of the participants are private [11]. We employ the concepts of game theory – particularly those of mechanism design in the design of incentives for encouraging resource contribution and efficient usage of these resources for OCCS platforms in the remainder of the sections of this paper.

A. The Modeling Process Methodology

The methodology for the modeling process is as follows: We begin the modeling process by defining key properties of the real world system that is to be modeled. Next we outline the desired properties that the incentive schemes should possess and hence the modeling process should achieve. We then create a model of the real world system and design the incentives schemes. We also review related work involving incentive design for cloud computing resource allocation and game theory and refine the model with some inspiration from the reviewed related work. We perform analysis to compare the results with the desired properties of the incentive mechanisms. Finally, we iteratively refine the model and the incentive mechanisms until the result approaches the desired properties.

The key properties of the OCCS platform to be modeled include the nature of the members and services, the credit points and benefits, knowledge of information on the platform, and the resource allocation.

1) Nature of Members and Services

The OCCS network consists of a set of strategic members contributing and utilizing cloud computing services. The platform consists of a set of services each belonging to a category; each service has a non-monetary cost that varies dynamically. The service or resource contributed by a member is of a certain finite capacity and the resources to a particular service may be contributed by multiple members. It should be noted that though there may be members that are supposedly altruistically offering services without directly benefiting from using services provided by others, they may still be doing it strategically since they may be doing it for example for advertisement purposes.

Members will normally only contribute resources that they have spare capacity of (e.g. CPU, storage, application that they have developed internally, etc.). We term these resources contributable resources. It should be obvious that a member's IT resources that it has need of using for its own production will constitute its non-contributable resources. Contributable resources incur a base personal (non OCCS platform) maintenance cost whether or not they are contributed; and also incur additional utilization cost when they have been contributed and are being utilized by other members. Members are free to provide and discontinue one or more services at will at any point in time. They are likewise free to use or discontinue the usage of one or more services at will at any point in time.

2) OCCS Platform credit Points and Benefits

Members are credited with OCCS platform points for contributing resources towards a service and debited OCCS platform points for resource usage. The only way to obtain credits on the platform is by accumulating the credit points through contributing resources to the platform. Members are interested in maximizing their OCCS platform credit points obtained for resource contribution and minimizing OCCS platform cost incurred in resource usage. Members receive a certain personal (non OCCS platform) benefit from utilizing a service; e.g. the benefit they obtain in using the service for providing services to their customers or in some cases simply as a form of advertisement.

3) Knowledge of Information on Platform

Members are aware of the activities of other players, and actions are sequential. Since the history about a service is known only to a certain extent they may or may not have perfect information about a particular service they might be interested in.

4) Resource Allocation

When the demand on a service is higher than its resources can support, interested users of the service with currently higher accumulated OCCS credit points are giving higher priority to use such resources. They may however choose not to use the service now and rather use it when the demand and hence the cost go down. However since their business processes may depend on using the service now in order to provide a certain level of quality of service (e.g. availability) to their customers, they may be constrained to use the service at peak demand. Members can be allocated recourses even when their OCCS credit is zero or negative.

B. Desired Features of the Incentive Schemes

The main features of the incentive schemes are as follows: It needs to be incentive compatible. That is it induces cooperation among an otherwise rational or selfish players. Secondly, it does not only encourage contribution of resources but also the efficient usage of these resources. Furthermore it must ensure the existence of sustainable equilibrium under normal operation and also ensure that the platform quickly arrives at a stable equilibrium when genuine disturbances take place (e.g. discontinued services or increase in demand for a particular service). Additionally, it must be robust enough to prevent or foil the effect of malicious attacks on the incentive scheme either by members on the OCCS platform or external entities. The collection of incentive scheme must also be flexible enough to adapt to changing user behavior on the platform. Ideally the individual schemes must be modular so that a combination of them can be applied simultaneously to achieve specific effects depending on the observed characteristics of their effects on the platform. This is important so that it will not be necessary to continually call for the redesign of the incentive scheme.

C. Model of Incentive Schemes for OCCS

This section begins with a detail description of the system model. It is followed by the adopted notations together with the utility payoff matrix that will be used throughout the rest of the paper. Two models of the type of games namely non cooperative and cooperatives games are presented and the existence of pure strategy Nash-Equilibrium shown for them.

1) System Model

Each service or group of substitutable services on the platform constitutes a separate game. Multiple games are being played simultaneously. A member participates in a particular game by either contributing resources to the service, using resources of the service, or express interest in the service by subscribing to it. A user may play in multiple games at the same time. We however assume players resource contribution or usage on one service is not dependent on that of other services and hence the games are independent. Their strategies in a particular game are therefore independent of those in others being played concurrently.

Service contribution and utilization are continuous in time. We however make analysis on an arbitrary time slot (e.g. 1hr, 1day etc. depending on the nature of the service and the corresponding billing model). The games are thus inherently repeated games. The time horizon consist of the active slot period corresponding to the time slot for which analysis and payoffs are going to be made, and the preparation period. All time slots preceding an active slot and has not already expired are potentially part of the preparation period for that active slot. Resources contributed towards a service during a particular slot are immediately available in the next time slot unless otherwise specified by the contributor when the resources will be available, in which case the OCCS platform does not consider it a resource until the specified time. Potential users of a service explicitly specify when they want to use the service and hence the active slot(s) they are interested in.

2) *Notation*

We have a nonempty, finite set I of $n \in \mathbb{N} \equiv \{1, 2, 3, \ldots\}$ players in a game. Contributable resources incur a base maintenance cost of m_i whether or not they are contributed and also incur additional utilization cost u_i when they have been contributed and are being utilized by other players. Players receive a certain benefit b_i from utilizing a service e.g. the benefit they obtain in using the service for providing services to their customers. To clarify the various costs involved in the discussion of the notations above we use a simplified scenario where player **A** has contributable resources to a service for which player **B** has need of and hence is a potential user of this resource. If player **A** chooses to contribute the resource, then resource R_a is available to the OCCS platform which if player **B** decides to use will pay a cost C to the platform of which αC is credited to player **A**. In this case player **A** also incurs a utilization cost u_a in addition to the maintenance cost m_a while player **B** derives a benefit b_b from the usage of this resource.

Table 1 show the payoffs $\mu(s)$ on the OCCS platform where the row player is assumed to have a contributable resource while column player has need of such a resource. This scenario is also depicted in figure 1. The payoffs in the table are for per unit capacity contributed or used. Thus the actual credit points φ_i obtained or debit incurred on a particular service (game) during the active slot under consideration is the product of per unit payoff and the quantity of resource contributed toward or utilized on that service. $\varphi_i(s) = q_i \; \mu_i(s)$ where q_i , is the quantity contributed or requested for use by player i and $\mu_i(s)$ is it's per unit payoff. The **Off** state in Table 1 for the row player indicates that though it has contributable resources, it chooses not to offer them to the OCCS platform for the active time slot under consideration. Similarly the **Off** state for the column player indicates that though it has need of that particular resource it chooses not to use this service in this active time slot under consideration. It can be seen from table 1 that resource contributors are only awarded OCCS credit points when their resources have been utilized by others. This will ensure that resource contributors are encouraged to contribute only resources that other users on the platform will find useful to use thereby encouraging cooperation between the contributors of resources and the corresponding potential users of these resources.

The role of parameter α is to determine the importance (1-α) attached to the administration of the OCCS governance platform itself. The resources required for the management of the OCCS platform can be viewed in one of two ways. One view will be to treat it as any other service on the platform in which case α can be set to 1; on the other hand it could be argued that since the whole concept collapses without the OCCS platform then it is a critical service which needs special attention.

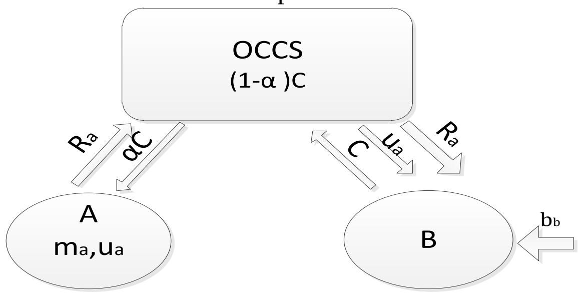

Figure 1: Scenario of member A having a contributable resource to a service of which player B have need of and is therefore a potential user of this resource.

TABLE 1: UTILITY PAYOFFS ON OCCS PLATFORM

Strategies	**Use**	**Off**
Supply	αC - m - u, b - C	- m,0
Off	- m,0	- m, 0

Row player is assumed to have a contributable resource while column player has need of the service. $0 < \alpha \leq 1$

The strategy s_i of player i is a non-negative real valued function of the quantity q_i that the player chooses to supply or request to use and the reported type $\hat{\theta}_i$ of its true type $\theta_i \in \Theta_i \rightarrow \mathbb{R}_+$ .The set of types θ of all players is given by $\theta = (\theta_1, \theta_2, \ldots \theta_I)$ where

$$\theta_i = \begin{cases} m_i + u_i \;,\; i \in supply \\ \\ b_i \;,\; i \in use \end{cases}$$

$$s_i = (q_i, \hat{\theta}_i) \rightarrow \mathbb{R}_+^2 \qquad (1)$$

3) Game Models

Two types of game models are considered; the non-cooperative game and the cooperative game models.

The goal of a player in the non-cooperative game model is to maximize only its own profit.

$$max. q_i \mu_i(s) \ \ s.t. \ q_i \leq K_i, \ K_i \text{ is the capacity of } i \ . \qquad (2)$$

In the cooperative game, the goal of the player is to maximize its own profit together with that of its collaborating partners $P(i)$.

$$max. \ q_i \mu_i(s) + \sum_{j \in P(i)} q_j \mu_j(s) \ \ s.t. \ \ q_i \leq K_i$$

$$and \ \forall \ j, q_j \leq K_j \qquad (3)$$

Though the OCCS platform does not require members to explicitly establish ties with other members, over time such ties may evolve due to the incentive compatibleness of the incentive schemes as is discussed later, whereby player i identifies key members on the platform whose strategies has significant effect on its own payoffs and therefore will want to guard their interest. The relationship in this case in not necessarily mutual, that is for two players i and j

$$j \in \ P(i) \not\Rightarrow i \in \ P(j)$$

A second possibility on the other hand is the scenario where two or more members decide to collaborate for their collective interest; these could be a group of contributors or resource users of a service or a mixture of both. In this case

$$j \in \ P(i) \Rightarrow i \in \ P(j)$$

4) Existence of Equilibrium

The strategy s_i of player i is a non-negative real valued function of the quantity q_i that the player chooses to supply or request to use and the reported type $\hat{\theta}_i$ of its true type $\theta_i \in \ \Theta_i \ \rightarrow \mathbb{R}_+$

$$s_i = \left(\ q_i, \theta_i \right) \rightarrow \mathbb{R}_+^2$$

This lies within the closed ball $\mathcal{B}_i \subset \mathbb{R}_+^2$ which is defined by $q_i \leq \ K_i \ , \ \theta_i \in \ \Theta_i \leq \ \Theta_{ki}$

We now define s_{-i} as the strategies of all other players other than i. This is a vector of the individual strategies $s_{i'} \ \ i' \neq i$. We now formulate *theorem1* which shows the existence of a Nash-equilibrium.

Theorem1: for both the non-cooperative and the cooperative game models there exists at least one pure strategy Nash equilibrium, i.e. a set of strategies $\bar{\boldsymbol{s}}_{\boldsymbol{i}}$ such that

$$\varphi_{\boldsymbol{i}}\left(\bar{\boldsymbol{s}}_{\boldsymbol{i}}, \bar{\boldsymbol{s}}_{\boldsymbol{-i}}\right) \geq \varphi_{\boldsymbol{i}}\left(\boldsymbol{s}_{\boldsymbol{i}}, \bar{\boldsymbol{s}}_{\boldsymbol{-i}}\right) \ \ for \ \ i \in I, \ \boldsymbol{s}_{\boldsymbol{i}} \in \mathcal{B}_i$$

Proof: The proof of the theorem is based on Kakutani's fixed point theorem [12] which has been used by a number of authors in the proof of the existence of Nash-equilibrium [13], [14]. The proof is shown in the Appendix1.

D. Incentive Schemes

From the perspective of the OCCS platform to ensure the sustainability of the network through encouraging contribution of resources and the efficient utilization of these resources, the objective is to maximize the total utility of every service on the platform subject to the constraints that for each service the utilized resources are less or equal to the contributed resources. The social choice function $f : (S_1, \Theta_1) x (S_2, \Theta_2) \ldots x (S_I, \Theta_I) \ \rightarrow O$ choses an outcome $f(s, \theta) \in O$ given the types $\theta = (\theta_1, \theta_2, \ldots \theta_I)$

$$f(s,\theta) = max. \sum_{i=1}^{n} \varphi_i = max. \sum_{i=1, \ i \in supply}^{n} q_i \mu_i(s,\theta) + \sum_{i=1, \ i \in use}^{n} q_i \mu_i(s,\theta)$$

$$s.t. \ q_i \leq \ K_i \ \ \forall \ i$$

$$and \ \ L = \sum_{i=1, \ i \in use}^{n} q_i \ \leq \sum_{i=1, \ i \in supply}^{n} q_i = G \qquad (4)$$

Mechanism design is the sub field of microeconomics and game theory that considers how to implement good system-wide solutions to problems that involve multiple self-interested agents each with private information about their preferences. A social choice is an aggregation of the preferences of the different participants towards a single joint decision which is literally the common good of the participants as a whole in contrast to their individual interests. So in our case on the OCCS platform, the social choice is to have resources offered by members to the platform and have these resources used efficiently (not abused by other members). The social choice function $f(s, \theta)$ is therefore to maximize the total utility payoff of all services on the OCCS platform which will ensure that the desired social choice is achieved.

For notational simplicity we drop θ as $s_i = (q_i, \hat{\theta}_i)$ and the reported type θ_i is a function of agent i's actual type θ_i . Equation (4) is rewritten to be

$$f(s) = max. \sum_{i=1, \ i \in supply}^{n} q_i \mu_i(s) + \sum_{i=1, \ i \in use}^{n} q_i \mu_i(s)$$

$$s.t. \ q_i \leq \ K_i \ \forall \ i \qquad (4b)$$

This reduces to an optimization problem in which the cost C that maximizes the total utility payoff of all the players is computed and then resources allocated. Equation (4) is a combination of linear functions and any suitable optimization algorithm such as Linear Programming can be applied in solving it.

1) Variations of the Base Scheme

The scheme in equation (4) is in our context the simplest form of incentive scheme which is based on the utility payoffs in table 1 that ensures that service resource contributors only get credit points when their resources have been used by interested members. This will form the base scheme. This section will present variations of this base scheme that add new constraints to it that seek to achieve specific features of the incentive mechanism. These new schemes are the Dominant Strategy Scheme (DSS), Equi-Profit Scheme (EPS) and Dominant Equi-Profit Scheme (DEPS).

a) Dominant Strategy Scheme

This scheme introduces new constraints on the base scheme in (4) so that the set of contributors collectively and the set of utilizers of the resources collectively have dominant strategies. In game theoretic terms a player is said to have a dominant strategy $s_i' \in S_i$ if s_i' gives player i a higher expected utility payoff than does every other $s_i \in S_i$ for every possible deleted pure strategy profile $\boldsymbol{s}_{-i} \in \boldsymbol{S}_{-i}$ which the opponents could play. That is if player i has contributable resources it will always choose to contribute if $\alpha C - u_i \geq 0$; and will always use resources if $b_i - C \geq 0$ when it has need of such a service. The dominant strategy scheme's constraint therefore is that

$$\frac{\sum_{i=1,i\in supply}^{n} q_i\left(\alpha C - u_i\right)}{G} \geq 0 \text{ and } \frac{\sum_{i=1,i\in use}^{n} q_i(b_i - C\ )}{L} \geq 0$$

(5)

It must be noted that equation (5) does not attempt to achieve a dominant strategy equilibrium in the classical game theoretic sense as all agents likely will not have a dominant strategy but rather it is the set of contributors collectively and the set of utilizers collectively that must have a dominant strategy.

b) Equi-Profit Scheme

The equal profit scheme also introduces a new constraint on the base scheme in (4). This constraint is that the average per unit utility payoff for contributors of resources is equal to that of the utilizers of these resources. Let

$$\varphi_u = \sum_{i=1,i\in use}^{n} q_i \mu_i(s) = \sum_{i=1,i\in use}^{n} q_i(b_i - C) \text{ and}$$

$$\varphi_s = \sum_{i=1,i\in supply}^{n} q_i \mu_i(s) = \sum_{i=1,i\in supply}^{n} q_i(\alpha C - m_i - u_i)$$

Then $$\frac{\varphi_s}{G} = \frac{\varphi_u}{L} \qquad (6)$$

The scheme seeks to ensure fairness towards both contributors of resources and their utilizers thereof, and thereby guarding against any single player or collaborating players of a particular category teaming up against the other category. Hence this should force collaboration between suppliers of the resources and the users of these resources.

c) Dominant Equi-Profit Scheme

This scheme combines both of the constraints of dominant strategy and the equi-profit schemes to the base scheme in (4) such that the set of contributors of resources and the set of utilizers of these resources have dominant strategy; and additionally the average per unit utility payoff is the same for both contributors and utilizers of the service.

2) Advanced Schemes

In addition to the variations in the base scheme, we have two schemes also targeted at achieving specific features in the desired properties of the incentive mechanism. However unlike the variations of the base scheme, they do not add constraints but apply game theory concepts in achieving this.

a) Discount Factor Scheme

This scheme introduces the consideration of a discount factor $\left(0 < \delta < 1\right)$ element to the variations of the base schemes. In game theory the discount factor denotes how much a future payoff is valued at the current period. On the OCCS platform it is also a measure of a player's perception of the continuity of both the platform and the services that it is interested in. For a contributor of resources to a particular service, δ_i is an indication of the value placed on OCCS credits points obtained now as compared with when it could make use of these credits points later on the platform when it is interested in using a service. On the other hand for a player currently using a service, it potentially may never need to provide services to compensate the platform for the debits incurred now. If it even does, it is similar to obtaining the service on high purchase.

The discount factor can be applied to any of the base schemes discussed above. φ_u and φ_s in (6) become

$$\varphi_u = \sum_{i=1,i\in use}^{n} q_i(b_i - \frac{C}{\delta_i}),$$

$$\varphi_s = \sum_{i=1,i\in supply}^{n} q_i(\alpha\delta_i C - m_i - u_i)$$

b) Stochastic Scheme

The stochastic scheme allows for flexibility in the specification of the maintenance cost m_i and utilization cost u_i for contributors of resources and the benefits b_i for the utilizers of resources. This scheme is useful in specifying these parameters as random variables for two reasons. The first is that utilizers of resources on the platform may be uncertain about the specific value they derive from the resource usage and contributors may also be uncertain about their maintenance and utilization costs. Secondly it may be needful for the OCCS platform to automatically predict these parameters based on the history of members on the platform and their resource contribution and utilization patterns. Thus in addition to offering flexibility in specifying these parameters it can also guard against members specifying arbitrary values for these parameters.

The stochastic scheme together with the discount factor scheme discussed above lays the foundation of our future work on performing repeated game analysis and dynamic mechanisms design for incentive schemes on the OCCS platform.

IV. Resource Allocation

The outcome o of a game is the resource allocation function $g(s)$ and the transfer function $t(s)$. The resource allocation function $g(s)$ allocates resources to resource demanders and assigns these allocations to contributions. The transfer function $t(s)$ credits OCCS points to resource contributors and debits utilizers of these resources according to the allocations made.

A. The Resource Allocation Process

The resource allocation process has an inbuilt robustness scheme that guard against the exploitation of the resource allocation process by utilizers of resources with transient membership – that is member with the intension of being on the platform for only a few active time slots. It must be noted that the application of game theory in modeling the incentive mechanisms would have naturally thwarted this had it not been that one of the desired features of the schemes as stated in Section III A.4 is that members can be allocated recourses even when their OCCS credit is zero or negative.

The resource allocation process redefines the set of resource demanders (***use)*** to be only genuine utilizers of resources. It applies various criteria in determining the genuine utilizers. For example a user could potentially exploit the resource allocation process by specifying arbitrarily high b_i thereby always getting resources allocated to it for which it may never have to later compensate the platform for. This is guarded against for example by requiring that reported b_i by a resource demander in a particular active time slot, be less than twice the average benefits $\bar{b}$ indicated by all utilizers of resources for this particular service during this active time slot, in order for it to be among the set of genuine users.

Resources are allocated to maximize the total allocated payoffs. First the amount of allocatable resources is computed as the minimum of the total supplied resources and the total demanded resources by those interested in using the service. The set of ***use*** is then sorted in descending OCCS accrued points and then by profitability $(b_i - C)$. Resources are then allocated while the amount of allocated resources is less than the allocatable limit. Next the set of contributors (***supply)*** is arranged in descending profitability $(\alpha C - m_i - u_i)$ and demands from resource utilizers assigned to them while the allocated supply is less than allocatable.

B. Evaluation of the Incentive Mechanisms

1) Budget Balance

We show weak budget balance of the mechanisms by showing that the net transfer to the mechanisms is non-negative at equilibrium $\sum_i t_i(s^*) \geq 0$

$$\sum_i t_i(s^*) = \sum_{i=1, i\in use}^{n} t_i(s^*) + \sum_{i=1, i\in supply}^{n} t_i(s^*)$$

$\sum_i t_i(s^*) = \sum_{i=1, i\in use}^{n} q_i^* C + \sum_{i=1, i\in supply}^{n} (-\alpha C q_i^*)$ where q_i^* is the quantity of resource allocated or assigned to a player i at equilibrium.

$$\sum_i t_i(s^*) = C\left(\sum_{i=1, i\in use}^{n} q_i^* + \sum_{i=1, i\in supply}^{n} (-\alpha q_i^*)\right)$$

But $\sum_{i=1, i\in use}^{n} q_i^* = \sum_{i=1, i\in supply}^{n} q_i^* = Q^*$ hence

$$\sum_i t_i(s^*) = CQ^*(1-\alpha) \geq 0 \text{ ,} since\, C \geq 0, Q^* \geq 0, 0 < \alpha \leq 1$$

2) Individual-Rationality

To show individual rationality [11] [15] [16] (actually ex-post individual rationality), we show that for all agents $i \in I$ the total utility φ_i^* of agent i in the equilibrium outcome of the mechanism is always greater or equal to the agent's utility $\hat{\varphi}_i$ for not participating in the mechanism. We first show for $i \in use \subset I$ then after we show for $i \in supply \subset I$.

For $i \in use, \hat{\varphi}_i = 0, \varphi_i^* = q_i^*(b_i - C^*)$

The allocation function $g(s)$ maximizes $\sum_{i=1}^{n} \varphi_i$ and decides on q_i^* based on $(b_i - C^*)$. $g(s)$ sets

$$q_i^* = \begin{cases} q_i^* \geq 0, if\,(b_i - C^*) \geq 0 \\ q_i^* = 0, if\,(b_i - C^*) < 0 \end{cases} \quad hence\, \varphi_i^* \geq 0$$

Similarly,

for $i \in supply, \hat{\varphi}_i = -m_i, \varphi_i^* = q_i^*(\alpha C^* - m_i - u_i)$,

$g(s)$ sets

$$q_i^* = \begin{cases} q_i^* \geq 0, if\,(\alpha C^* - m_i - u_i) \geq 0 \\ q_i^* = 0, if\,(\alpha C^* - m_i - u_i) < 0 \end{cases} \quad hence\, \varphi_i^* \geq 0 > \hat{\varphi}_i$$

3) Incentive Compatibility and Allocative Efficiency

In an incentive compatible mechanism the equilibrium strategy profile $s^* = (s_1^*, s_2^*, \ldots s_I^*)$ has every agent reporting its true preference to the mechanism at equilibrium. We prove incentive compatibility [17] of our mechanisms by showing that truth revelation is equilibrium (a pure strategy Nash equilibrium) of the games induced by the mechanisms; and that the outcome rule $g(s)$ implements the social choice function $f(s)$.

Proof: We proved in section III.C.4 as is shown in appendix1 the existence of a pure strategy Nash equilibrium for both the non-cooperative and cooperative games induced in the mechanisms. In the mechanism implementations the allocation function $g(s)$ maximizes $\sum_{i=1}^{n} \varphi_i$ (and is therefore Allocative efficient) and decides on the allocated resource q_i' for resource demanders based on $(b_i - C)$ and the assigned resources to contributors q_i' based on $(\alpha C - m_i - u_i)$. Thus $g(s) = f(s)$; the outcome rule is precisely the social choice function.

V. Conclusion and Future Work

This paper has looked at the design of incentive schemes that encourage the contribution of resources to the OCCS platform as well as the efficient usage of such resources. Game theory has been employed to model and design the incentive schemes with two game models presented. The existence of a pure strategy Nash equilibrium for both the cooperative and non-cooperative games has been shown. Three base incentive schemes have also been presented. These schemes are the Dominant Strategy Scheme (DSS), Equi-Profit Scheme (EPS) and Dominant Equi-Profit Scheme (DEPS). We performed analytical evaluation of our incentive schemes and conclude that the schemes meet the desired properties of budget-balance, ex-post individual rationality, incentive compatibility, allocative efficiency, robustness, and flexibility to accommodate changing user behavior on the platform.

Though these incentive schemes have been designed for the OCCS platform, they can also be applicable to a general incentive and resource allocation problem in which the service contributors will be one or more commercial cloud service providers servicing a collection of clients with their spare capacities on which they put no fixed price tag.

As was stated in the systems model in Section III.C.1, the service contribution and utilization are continuous in time. We however made analysis on a single time slot for which payoffs are to be made. Since the real world system induces inherently repeated games, the stochastic scheme and the discount factor scheme discussed in Section III.D.2 lay the foundation of our future work on performing repeated game analysis and dynamic mechanisms design for incentive schemes on the OCCS platform. We will also consider these further works in the context of a decentralized OCCS platform.

Finally, this work has provided the foundation for the implementation of Pseudo Service Level Agreement (Pseudo SLA) on the OCCS platform which is discussed later in the sequel to this paper on “Trust engineering and Pseudo SLA in Opportunistic Cloud Computing Services”. It is evident from the title of the sequel to this paper that we consider data privacy, security and trust as major issues that OCCS and cloud computing in general has to address. We are therefore currently working on trust engineering in cloud computing and how to adapt it for the OCCS environment.

References

[1] Fernando Costa, Luis Silva, and Michael Dahlin, "Volunteer Cloud Computing: MapReduce over the Internet," in *2011 IEEE International Parallel & Distributed Processing Symposium*, 2011, pp. 1855-1862.

[2] Eric Kuada and Henning Olesen, "A social network approach to provisioning and management of cloud computing services for enterprises," in *CLOUD COMPUTING 2011 : The Second International Conference on Cloud Computing, GRIDs, and Virtualization*, Rome, Italy, Sep.2011, 2011, pp. 98 - 104.

[3] Martin J Osborne and Ariel Rubinstein, *A course in game theory*, 1st ed. Cambridge, Massachusetts, London, England: Massachusetts Institute of Technology Press, 1994.

[4] Kevin Leyton-Brown and Yoav Shoham, *Essentials of game theory: A concise, multidisciplinary introduction*, Ronald J Brachman and Tom Dietterich, Eds.: Morgan & Claypool Publishers, 2008. [Online]. http://www.morganclaypool.com/doi/pdf/10.2200/S00108ED1V01Y200802AIM003

[5] Eric Kuada, Henning Olesen, and Anders Henten, "Public policy and regulatory implications for the implementation of opportunistic cloud computing services for enterprises," in *9th International Workshop on Security in Information Systems*, Wroclaw, 2012, pp. 3-13, http://www.iceis.org.

[6] Wayne Jansen and Timothy Grance, "Guidelines on security and privacy in public cloud computing," Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, Draft Special Publication 800-144 800-144, 2011.

[7] Danilo Ardagna, Barbara Panicucci, and Mauro Passacantando, "A game theoretic formulation of the service provisioning problem in cloud systems," in *WWW 2011 – Session: Monetization II*, Hyderabad, India, 2011. [Online]. http://www.www2011india.com/proceeding/proceedings/p177.pdf

[8] Zhen Kong, Cheng-Zhong Xu, and Minyi Guo, "Mechanism design for stochastic virtual resource allocation in non-cooperative cloud systems," in *2011 IEEE 4th International Conference on Cloud Computing*, 2011, pp. 614 - 621. [Online]. http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=06008762

[9] Guiyi Wei, Athanasios V Vasilakos, Yao Zheng, and Naixue Xiong, "A game-theoretic method of fair resource allocation for cloud computing services," *Springer*, vol. J Supercomput, no. 54, pp. 252 - 269, 2010.

[10] Chiranjeeb Buragohain, Divyakant Agrawal, and Subhash Suri, "A Game Theoretic Framework for Incentives in P2P Systems," in *Third International Conference on Peer-to-Peer Computing (P2P 2003)*, 2003, pp. 48-56.

[11] Noam Nisan, "Introduction to mechanism design (for computer scientists)," in *Algorithmic Game Theory*. New York, USA: Cambridge University Press, 2007, ch. 9, pp. 209-241, 978-0-521-87282-9.

[12] Shizuo Kakutani, "A generalization of Brouwer's fixed point theorem," *Duke Mathematical Journal*, vol. 8, no. 3, pp. 457-459, 1941.

[13] John F Nash, "Equilibrium points in n-person games," in *Proceedings of the National Academy of Science(NAS)*, vol. 36, 1950, pp. 48-49.

[14] Juan Pablo Torres-Martinez, "Fixed points as Nash equilibria," *Fixed Point Theory and Applications*, vol. 2006, pp. 1-4, October 2006, Article ID 36135.

[15] David C Parkes, "Online mechanisms," in *Algorithmic Game Theory*. New York, USA: Cambridge University Press, 2007, ch. 16, pp. 411-439.

[16] Danish Garg, Y Narahari, and Sujit Gujar, "Foundations of mechanism design: A tutorial Part 2 – Advanced concepts and results," *Sadhana, Indian Academy Proceedings in Engineering Sciences*, vol. 33, no. 2, pp. 131-174, April 2008.

[17] Matthew O Jackson, "Mechanism theory," in *Optimization and Operations Research*, Ulrich Derigs, Ed. Oxford, UK: Encyclopedia of Life Support Systems (EOLSS), EOLSS Publishers, 2003, ch. 4, Developed under the Auspices of the UNESCO, http://www.eolss.net.

[18] Gyorgy Dan, "Cache-to-Cache: could ISPs cooperate to decrease peer-to-peer content distribution costs?," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 9, pp. 1469-1482, September 2011, http://doi.ieeecomputersociety.org/10.1109/TPDS.2011.28.

APPENDICES

A. *Appendix1*

Lemma 2 (Kakutani): Let $\mathcal{B} \subseteq \mathbb{R}^{|\mathcal{H}|}$, $\mathcal{B}$ compact, convex and non-empty. Let $\mathcal{K}$: $\mathcal{B} \rightrightarrows \mathcal{B}$ be a correspondence (non-empty valued). Such that $\mathcal{K}$(b) is convex ∀ b∈$\mathcal{B}$. assume moreover that $\mathcal{K}$ has closed reduced graph. Then there is a fixed point for $\mathcal{K}$, i.e. $\exists b \in \mathcal{B}$ s.t. $b \in \mathcal{K}\left(b\right)$ [12]. The following proof of Theorem 1 consists of showing that the conditions of Lemma 2 are satisfied. The proof is based on [18] which is also based on [13].

Proof: (**Theorem 1**) $\mathcal{B}_i$ is non-empty because for $\mathcal{K}_i > 0$ there is at least one feasible strategy. $\mathcal{B}_i$ is closed and bounded, hence it is compact. Furthermore, $\mathcal{B}_i$ is convex due to the capacity constraint being linear (2). The payoff function that player i tries to maximize is continuous in q_i and $\mu_i\left(s\right)$ in both the non-cooperative game (2) and the cooperative game (3). And it is quasi-concave as it is linear. We define the set valued best response function of player i.

$$\mathcal{K}_i(s_{-i}) = \left\{s_i \in \mathcal{B}_i | \varphi_i(s_i, s_{(-i)}) \geq \varphi_i(s_i', s_{(-i)}) \forall s_i \in \mathcal{B}_i\right\}$$

The set $s_i = \mathcal{K}(s_i)$ is non-empty because φ_i is continuous and $\mathcal{B}_i$ is compact. It is convex due to the quasi-concavity of the payoff function φ_i . The graph of $\mathcal{K}_i$ is closed because of the payoff functions are concave.

Let us define $\mathcal{B} = \text{x}_{i\in I}\ \mathcal{B}_i$ and the correspondence $\mathcal{K}$: $\mathcal{B} \rightrightarrows \mathcal{B}$ as $\mathcal{K} = \text{x}_{i\in I}\ \mathcal{K}_i$. $\mathcal{B}$ is hence compact, convex and non-empty, and $\mathcal{K}$ is convex, non-empty valued and has closed reduced graph. Hence, due to Kakutani's theorem $\mathcal{K}$ has a fixed point such that $s_i = \mathcal{K}(s_i)$, which proves the existence of a Nash-equilibrium both for the non-cooperative and the cooperative strategies.