Auditing and Provision Strategies of Utility Computing Services: A Game Theoretic Perspective

Yung-Ming Li

Institute of Information Management, National Chiao Tung University

Yi-Lin Lee

Institute of Information Management, National Chiao Tung University

Abstract

The emergence of utility computing promises to transform the way corporations possess their IT infrastructure. This model of computing may shift IT infrastructure from being a fragmented capital asset to being a centralized utility service. As a result of increasingly competitive business environment, organizations are looking for new approaches to minimize IT costs. This forces organizations to focus on their core business and outsource computing demand. Past researches about utility computing mainly concentrate on the pricing strategy and resource utilization problem. In our paper the service broker is introduced into utility computing environment, and best response functions under mixed strategy for service broker and provider is analyzed. Our analytical results show that the service broker can dominate by controlling reward, penalty and service fee paid to service provider. In addition, in contrast to penalty, reward is more helpful to ensure the QoS. Reward, penalty and opportunity cost help to make decisions about audit and deviate in mixed strategy condition, while reward, opportunity cost and transition cost of service broker do the same thing in pure strategy.

Key words: Utility Computing, Competition, Incentive, Game Theory

以賽局理論探討公用運算之稽核機制與服務提供策略

李永銘

交通大學資訊管理研究所

李易霖

交通大學資訊管理研究所

摘要

　　公用運算架構的興起，改變了企業對於是否自己建置資訊科技基礎建設的態度，此一改變將使資源逐漸開始集中。面對外在競爭環境，企業開始思考如何專注於本業，並以尋求外部資源之方式有效降低龐大的資訊科技投資。以往有關公用運算架構的研究，多半集中於訂價策略與資源最佳化配置問題，本論文提出了一個新的架構，以服務中介者作為使用者與運算服務提供者的橋樑。在此架構下，我們由賽局理論觀點探討混合策略下中介者與運算服務提供者對於稽核機制與服務提供策略的最適反應。本研究發現中介者可藉由支付給服務提供者的獎賞、罰款以及服務使用費，來主導服務提供者的服務意願。與罰款機制相較，獎賞對於整個架構的運作是較為有幫助的。除討論混合策略外，單純策略之中的決策變數也一並被討論。在混合策略情況下，獎賞、罰款與機會成本會影響稽核機制與提供服務意願；在單純策略的情況下，則需考慮獎賞、中介者本身之服務轉接成本以及機會成本。

關鍵字：公用運算，競爭，誘因，賽局理論

1. INTRODUCTION

The growing of the Internet, powerful computers and high-speed networks as low-cost commodity components are changing the way we do computing. These technological developments have led to the possibility of using networks of computers as a single, unified computing resource (Pfister 1998; Werder 2004). The reliable and low-cost availability of data center services has encouraged many businesses to outsource their computing needs; thus heralding utility computing model. Utility computing is one service provision model that can be classified as an on-demand computing model (Vorisek & Feuerlicht 2004; Lodi et al. 2007). It presents a paradigm where shared infrastructure can be provided on demand to multiple applications and customers(Machiraju et al. 2002; Machiraju et al. 2004). Also, it is envisioned to be the next generation of information technology evolution that depicts how computing needs of users can be fulfilled in the future IT industry (Werder 2004). Its analogy is derived from the real world where service providers maintain and supply utility services. Consumers in turn pay service providers based on their usage(Yeo et al. 2006). This is different with the conventional computing model, that is, customers do not have to invest in owning resources anymore. As the utility computing provider can spread the customers' variance in resource needs, the utilization of the resources can be optimized. IDC expects the global market for utility computing offerings to cross USD 1 billion in 2004 from USD 0.5 billion in 2003 and to reach USD 4.6 billion in 2007. According to an IDC survey, outsourced utility computing will have a dramatic impact on a broad range of services industries, ranging from the IT and business process services markets to the communications market.

Utility computing deliver information services when needed, in such a way that customers neither incur the high fixed costs of purchasing hardware and software, nor commit to long-term fixed-price outsourcing contracts. Providers of utility computing must understand specific service needs and requirements of users in order to design suitable policies for them. With the changing demand of service needs from users, providers must be able to fulfill the dynamic fluctuation of peak and non-peak service demands. Uncharacterized, no guaranteed information resources are valueless (Chris & Giorgos 2004). Clients bear the risk of a supplier going out of business, or not being able to deliver on promised service levels when obtaining IT resources from outside providers(Earl 1996).

A utility computing service level agreements (SLAs) is an IT service contract that specifies the minimum expectations and obligations that exist between the provider and the customer of utility computing service (Buco et al. 2004). If the expected level of quality of service (QoS) is not met, providers will then be liable for compensation and may incur

heavy losses. Therefore, providers seek to maximize customer satisfaction by meeting service needs and minimize the risk of SLA violations (Buco et al. 2004). However, it may be too costly for service providers to build up the computing capacity to satisfy the peak need. In order to minimize the risk and reduce facilities investment, aligning with or outsourcing to other providers under specific situations is a feasible solution.

Currently, most of research concerning about utility computing mainly focus on the pricing strategy and resource utilization problem, and most of them assume that there are only service requestors and providers in utility computing. As we mentioned, user demands vary from time to time, thus it is hard for single service provider to fulfill these requirements all by self-owned facilities. In this paper we introduce a service broker acting as resource allocator and coordinator in utility computing. The broker can decide whether to outsource the user requests, under this situation the QoS guarantee issues arise. We try to propose an analytic model with audit mechanism for this new architecture, and the probability of not willing to supply required services by providers is also considered.

2. RELATED LITERATURE

2.1 Utility Computing

The word utility is used to make an analogy to other services, such as electrical power, that seek to meet fluctuating customer needs, and charge for the resources based on usage rather than on a flat-rate basis. This approach is becoming more common in enterprise computing and is sometimes used for the consumer market as well, for Internet service, Web site access, file sharing, and other applications. An utility computing model offers growing businesses access to a managed system which requires no up-front capital investment by the customer. A utility environment is dynamic in nature. It ensures the smooth operation of the supported services by dynamically adjusting the allocation of resources (Eilam et al. 2004). Utility computing presupposes that diverse computational resources can be brought together on demand and that computations can be realized depending on demand and service load (Huhns & Singh 2005). It has to deal with a large number of resources of varied types, as well as multiple combinations of those resources (Sahai et al. 2003). Resources actions are taken to achieve a desired quality of service (QoS). QoS goals are described in Service Level Agreements (SLAs) that consists of constraints placed on service related metrics (Hellerstein et al. 2005). And SLAs are a sine qua non in the deployment of utility computing (Farrell et al. 2004). In order to maintain SLAs, penalty mechanism is usually introduced (Yeo & Buyya 2005).

The utility computing model offers many benefits to both service providers and users. From the provider's point of view, actual computing resources are not set up to satisfy a single user. Instead, virtualized resources are created and assigned dynamically to various users when needed. Providers can reallocate resources dynamically to users that have the highest demands. This efficient usage of resources minimizes operational costs for providers since they are now able to serve a larger community of users without letting unused resources go unutilized. Utility computing also enables providers to achieve a higher Return on Investment (ROI) since shorter time periods are now required to derive positive returns and incremental profits can be earned with the gradual expansion of infrastructure that grows with user demands.

From users' perspective, the most important advantage of utility computing is the reduction of IT-related operational costs and complexities (Yeo et al. 2006). Users no longer need to invest heavily or encounter difficulties in building and maintaining IT infrastructures. Users neither need to be concerned about possible over- or under-utilization of their self-owned IT infrastructures during peak or non-peak usage periods, nor worry about being confined to any specific vendor's proprietary technologies. With utility computing, users can obtain appropriate amounts of computing power from providers dynamically based on their service needs and requirements. This is particularly useful for users who experience rapidly increasing or unpredictable computing needs. Such an outsourcing model thus provides increased flexibility and ease for users to adapt to their changing business needs and environments (Ross & Westerman 2004).

Utility computing promises the following benefits (Chang et al. 2004):

- Simplify IT by reducing complexity
- Turn IT from a fixed to a variable cost
- Reduce cost or operating expense

  The main benefits of the utility computing model for service providers are (Mani 2007):

- The computing service provider need not set up actual hardware and software components to satisfy a single solution or user, as in the case of traditional computing.
- Providers can reallocate resources with ease by the use of virtualized resources, which can be created and assigned dynamically to various users when needed.
- The operational costs for providers are reduced due to better resource utilization. The TCO is also reduced.

  While utility computing services deliver distinct benefits to customers of IT services, they pose new challenges for providers (Paleologo 2004):

- Reduced contract duration. Contracts duration could be further reduced in the future. In the new model, the challenge is associated with a portfolio of contracts,

and one uncertainty faced by the provider lies in the variable duration of these contracts.

● Reduced switching costs and customer lock-in. Set-up and fixed recurring fees constitute just a smaller percentage of the cumulative revenue for providers. In turn, this facilitates the migration of customers among providers.

● Uncertain customer demand. The core of the realized revenue is proportional to customer demand. If the customer base is sufficiently large, the change of customer demand have less impact on the profitability of the offering, and the provider only faces the risks associated with the industry sector in which the customers operate.

● Short life cycles and high sunk costs. The short contract duration allows customers to switch to the newest available technology at little or no cost because of lower switching costs. Sunk costs include development costs for instrumentation, provisioning, and monitoring of new services. Within the cost structure of utility-computing service offerings, sunk costs are much larger than the variable costs.

In order to have a clear understanding, utility computing is defined throughout this paper as:

Utility Computing is a service-provisioning model, in which service provider provisions IT services in infrastructure, application and business process areas on an on-demand basis and charges on usage basis.

2.2 The Roles of Service Broker

According to Webster's Dictionary, service refers to "useful labor that does not produce a tangible commodity". Further characteristics of services are immediacy, high customer involvement and difficulty of standardization (Werder 2004). Webster's dictionary defines broker as" an agent employed to effect bargains and contracts, as a middleman or negotiator, between other persons, for a compensation commonly called brokerage". The middleman between the involved actors can take up various roles depending on the area of contracts. In this paper, we define broker as service brokers playing in the service market. In a service market, a broker represents an intermediating function that provides information about providers, services, standards and contacts (Sieber & Griese 1999).

Valid services architecture requires at least the following three roles (Aphrodite & Thomi 2002):

● Service provider. A service provider is the party that provides software applications for specific needs as services. Service providers publish, unpublished and update their services so that they are available on the Internet. From a business perspective,

this is the owner of the service. From an architectural perspective, this is the platform that holds the implementation of the service.

- ● Service requester. A requester is the party that has a need that can be fulfilled by a service available on the Internet. From a business perspective, this is the business that requires certain functions to be fulfilled. From an architectural perspective, this is the application that is looking for and invoking a service. A requester could be a human user accessing the service through a desktop or a wireless browser; it could be an application program; or it could be another web service. A requester finds the required services via a service broker and binds to services via the service provider.
- ● Service broker. This party provides a searchable repository of service descriptions where service providers publish their services and service requesters find services and obtain binding information for these services. It is like telephone yellow pages. Utility computing architecture like Superglue also introduced service broker to its model (Rappa 2004).

The roles of service broker are illustrated in figure 1, and the interactions of the above roles are summarized in figure 2.
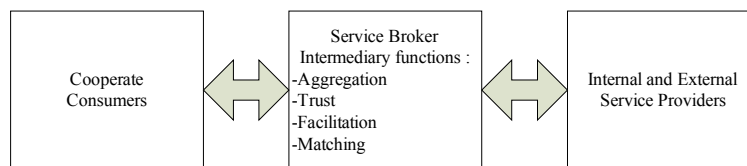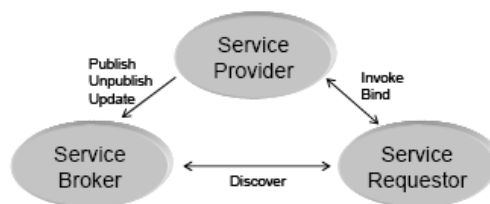

Figure 1: Service Broker (Bailey & Bakos 1997)

Figure 2: Basic Service Model (Aphrodite & Thomi 2002)

2.3 Service Pricing

Price has three functions in a market from a macro-economic perspective, namely allocation or rationing; stimulation and acting as an incentive for new players and products

to enter a marketplace; and distribution, whereby income is distributed between buyers and sellers (Rowley 1997).

Providers of utility computing are likely to price based on business value or willingness to pay, while their customers may be accustomed to thinking about paying for the infrastructure based on its cost (Huang & Sundararajan 2005). An inappropriate choice of pricing that is based on usage could either lead to excessive inertia in migration, or alternatively, to excess demand that providers cannot fulfill profitably or scale to meet reliably.

Paleologo (2004) proposed a methodology for pricing utility computing that takes risk into account, and reports on how it improves on simple cost-plus pricing models. Chen and Wu (2004) model a seller's choice of linear usage-based pricing for on-demand computing. Werder (2004) pointed out that increased perceived risk is one of the important factors that affect the pricing of IT-based services. The perceived risk of the customer is higher when purchasing a service than in the case of purchasing a product. This effect is often increased when dealing with IT-based services. Users who adopt utility computing services will evaluate the computing ability of service providers and make decision. However, all these models are focused on the charging mechanism for utility computing users, and the computing resources of service providers are assumed to be unlimited (Bhargava & Sundaresan 2004).

Huang and Sundararajan (2005) identify four aspects that affect utility computing pricing:

- The cost of buying, deploying and maintaining the infrastructure in-house
- The business value of the infrastructure
- The scale of the provider's infrastructure
- The variable costs like transitions, usages

Although this model has taken the computing resources of service provider into consideration, like other models, it also assumes that resources are unlimited.

2.4 Game Theory

Game theory has emerged recently as a powerful challenger to the conventional method of examining economics. The main purpose is to consider situations where their decisions are strategic reactions to other agents' actions. An agent is faced with a set of moves he can play and will form a best response strategy he will play. Game theory can be roughly divided into two broad areas: non-cooperative games and co-operative games. A cooperative game is a game where groups of players may enforce cooperative behavior; hence the game is a competition between coalitions of players, rather than between

individual players. A non-cooperative game is a one in which players can cooperate, but any cooperation must be self-enforcing.

Generally there are some assumptions in game theory (Rajeev & Heungsoo 1989; Eichberger 1993):

- Each player has two or more choices.
- Every possible combination of choices leads to an end-state which will terminate the game.
- Each decision maker has perfect information of the game; that is, he knows the rules of the game as well as the payoffs of all other players. All decision makers are rational, and each player will select the alternatives that yield him the greater payoff.

3. THE MODEL

It is common for firms to advertise performance promises, but these promises have little meaning when they are vague or enforceable or do not specify a penalty when promised performance is not delivered (Bhargava & Sundaresan 2004). In our model, utility computing service brokers act as resource allocator and coordinator. Its key performance indicators are usually related to service operations, service level agreements, or any other business indicators, and these are to be defined and audited(Aib et al. 2004). Utility computing is one of the ways to increase their return on investment in IT systems, in which business will contract third-party utility providers to provide IT services (Monahan 2005). If business relationships are based on the trustworthiness of third parties, it is crucial that these instances are subject to strict controlling and auditing regulations (Werder 2004). Thus for service broker to maintain QoS and trustworthiness between requestors and providers, an audit mechanism is required. Service providers provide computing resources on the command of service brokers. Service brokers provide a quality of service (QoS) guarantee to service requesters, while service providers also do the same thing. Self-owned computing resources can be facilitated by service brokers, and the QoS can be guaranteed. However, in order to meet the resource requirements at all time, hardware and software infrastructure has to be built and this leads to high sunk costs. The service broker can also acquire computing resources from external service providers. Self-owned facilities can provide stable and guaranteed service quality, but it is more costly than that of external service providers.
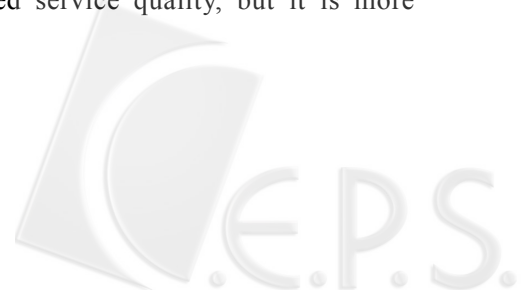
3.1 Contract Rules

For service providers, the computing resources can serve different service brokers. If the opportunity cost is higher than profit gained from service broker, the service provider will choose to violate QoS and reallocate resources to get more profit. In order to maintain QoS to service requestor, incentive is needed for service provider. The parameters used in our model are listed in table 1 and the rules of the contracting game are described below.

RULE 1: If the service broker meets the QoS guaranteed to service requestor, broker can charge price p; if QoS is violated, the requestor can get penalty b from broker without paying any service charge.

RULE 2: While receiving service request, the broker can dispatch to different service provider with service fee w. If the service provider cannot commit QoS, redispatching request with transition cost c is taken by the broker. Utility computing services require significant development and start-up costs because of demand uncertainty (Paleologo 2004). These are thus highly related to the estimation of user demand, and once invested it becomes sunk cost. In our model the transition cost is defined to be proportional to sunk cost and includes all the costs required to redispatch request except service fee to new service provider.

RULE 3: For service provider, the cost for fulfilling single service request is assumed to be k, and the opportunity cost for not providing service as advertised is v.

RULE 4: In order to maintain QoS agreement with service requestors, the incentive and penalty term for failure to deliver service levels stipulated in QoS has to be built (Chase et al. 2001). Audit is randomly taken and once the service provider supplies sufficient computing resources and pass the audit, a reward s will be received. If the broker audit while the provider violates QoS agreement, a penalty t is then received (transit from the service provider to the broker). The service provider will get nothing if he is audited and found to violate QoS agreement.

3.2. Payoff Functions

Denote θ the probability that the service provider violates QoS agreement with service broker and α the probability that the service broker will take the action of auditing the service offered by the service provider. Based on the assumptions and contract rules, the competition game between service broker and provider is described as following:
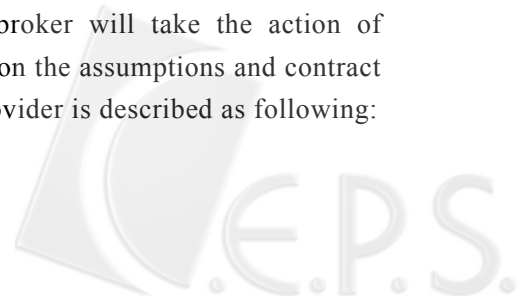
Table 1: Model Parameters

| Parameters	Description
p	the service charge price paid by service requestor to service broker if the QoS is met
b	the penalty paid by service broker to service requestor if the QoS is violated
w	the service fee paid by service broker to service provider while outsourcing service request to service providers
c	the transition cost of service broker
k	the cost of service provider to process single service request
v	the opportunity cost for not providing service as advertised
s	the reward for the service provider who supplies sufficient computing resources and pass the audit
t	the penalty for the service provider if the broker audit while the provider violates QoS agreement
θ	the probability that the service provider violates QoS agreement with service broker
α	the probability that the service broker will take the action of auditing the service offered by the service provider
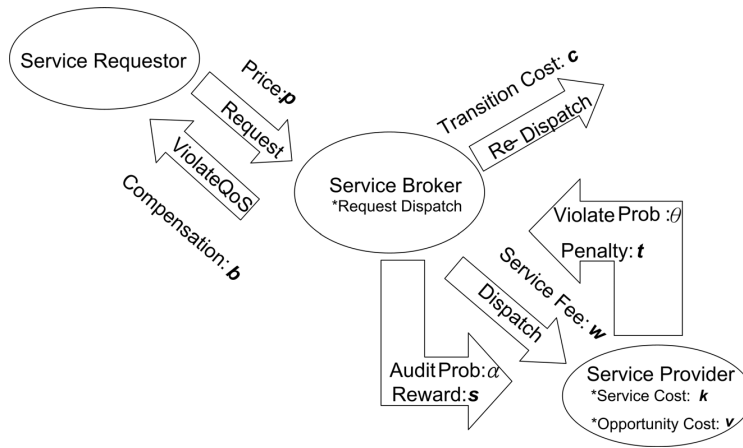
Figure 3: Service Architecture and Contract Rules

1. Service broker announces to service providers that a penalty will be placed if QoS agreement is violated. And in order to fulfill the service quality guarantee to service requestors, an audit action with probability α will be taken.

2. If the audit action is taken and the service provider passes, a reward will be received.

3. If the service broker cannot maintain the QoS because of resources shortage, a penalty has to be paid.

If the provider is audited and fails to pass, the penalty to provider is placed and the broker provides computing resources by himself. Under this condition:

Profit for service broker $\pi_B = (p+t)-c$,

Profit for service provider $\pi_P = v-(t+k)$.

However, if the service broker decides to audit and the provider passes, then:

Profit for service broker $\pi_B = p-(w+s)$,

Profit for service provider $\pi_P = w+s-(k+v)$.

Once the provider doesn't comply with QoS agreement and this is unknown to service broker, the broker has to pay the penalty to service requestor while receive nothing from provider. That is:

Profit for service broker $\pi_B = -b$,

Profit for service provider $\pi_P = (w+v)-k$.

If the service provider complies with QoS but is not audited, the broker pays ω to provider and gets service. But the provider won't get any reward since no audit action is taken.

Profit for service broker $\pi_B = p-w$,

Profit for service provider $\pi_P = w-(k+v)$.

The profits for service broker and provider are summarized as table 2.

3.3. The Game and Nash Equilibrium

The competitive scenario of the service broker and service provider can be formulated as a simultaneous game. We first investigate the mixed strategy Nash equilibrium results in which each of players stochastically select a strategy from his feasible strategy set (that is audit or don't audit for the service broker, and commit or don't commit for the service provider). Denote α the probability that the broker adopts audit action and θ the probability that the service provider adopts violate (i.e. don't commit) action.

Table 2: Profits for Service Broker and Provider

		Provider	
		Violates QoS (θ)	Not Violates QoS (1-θ)
Broker	Audit(α)	$(p+t)-c, v-(t+k)$	$p-(w+s), w+s-(k+v)$
	Not Audit(1-α)	$-b, (w+v)-k$	$p-w, w-(k+v)$

According to the payoff functions described, the objective for service broker and provider is to maximize the payoff. The expected payoff functions can be written as:

$$\max_{\alpha} E(\pi_B \mid \theta) = \alpha\big(\theta((p+t)-c)+(1-\theta)\big(p-(w+s)\big)\big)+(1-\alpha)\big(\theta(-b)+(1-\theta)(p-w)\big)$$

$$\max_{\theta} E(\pi_P \mid \alpha) = \theta\big(\alpha(v-(t+k))+(1-\alpha)((w+v)-k)\big)+(1-\theta)\big(\alpha(w+s-(k+v))+(1-\alpha)(w-(k+v))\big)$$ Compare the terms in both objective functions, we can find that the service broker's best response function to the service provider's provision strategy θ is

$$\alpha^*(\theta) = \begin{cases} 1 & \text{if } \theta > \hat{\theta} \\ [0,1] & \text{if } \theta = \hat{\theta} \\ 0 & \text{if } \theta < \hat{\theta} \end{cases}, \text{ where } \hat{\theta} = \frac{s}{b-c+p+s+t} \tag{1}$$

The service provider's best response function to the service broker's audit strategy α is

$$\theta^*(\alpha) = \begin{cases} 1 & \text{if } \alpha < \hat{\alpha} \\ [0,1] & \text{if } \alpha = \hat{\alpha} \\ 0 & \text{if } \alpha > \hat{\alpha} \end{cases}, \text{ where } \hat{\alpha} = \frac{2v}{s+t+w} \tag{2}$$

The relationships of these best response functions to the opponent's strategy are described as figure 4. We can easily observe that the dot point is the Nash equilibrium if mixed strategies are adopted. Obviously, Nash equilibrium results are associated with the parameter setting. Notice that the service broker has higher incentive to audit the service as the threshold $\hat{\theta}$ decreases (i.e. $\partial \mathrm{Pr}ob(\alpha^*=1)/\partial\hat{\theta}<0$), however, the service provider has higher incentive to deviate from QoS commitment as the threshold $\hat{\alpha}$ increases (i.e. $\partial \mathrm{Pr}ob(\theta^*=1)/\partial\hat{\alpha}>0$)
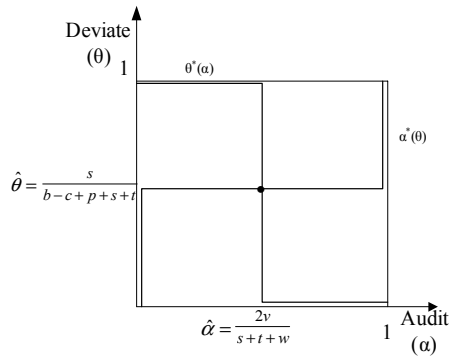

Figure 4: Best Response Function for Broker and Provider

PROPOSITION 1 (Mixed Strategy Nash Equilibrium). The optimal auditing and provision strategies is

$$\left(\alpha^*,\theta^*\right)=\left(\hat{\alpha},\hat{\theta}\right)=\left(\frac{2v}{s+t+w},\frac{s}{b-c+p+s+t}\right) \text{ where } 0<\alpha^*<1, 0<\theta^*<1 \tag{3}$$

and equilibrium expected profits for the providers are

$$(\pi_B^*,\pi_P^*)=\left(\frac{b(p-s-w)-(c-p-t)(p-w)}{b-c+p+s+t},w+v(\frac{2s}{s+t+w}-1)-k\right) \tag{4}$$

3.4. Analysis of Strategy Adoption

According to the derived mixed Nash equilibrium results, we have the following observation:

PROPOSITION 2 (Provision Strategy) The provision decision (QoS guarantee deviation intention) is (i) positively associated with contract reward and the transition cost of the service broker, (ii) but negatively with the price of the service, contract penalty, the penalty to the service requestor . That is $\partial \theta^* / y_1 > 0, where \ y_1 \in \{s, c\}$ *and* $\partial \theta^* / y_2 < 0, where \ y_2 \in \{b, p, t\}$.

We may imagine that the probability of deviation is highly related to the opportunity cost, however, our finding is opposite with this intuition. As we can observe from equation (1), all the decision variables in the threshold $\hat{\theta}$ are fully controlled by the service broker, but the opportunity cost doesn't show up. The service broker functions as the resource allocator in the utility computing, thus the provider will observe the incentives from broker and make decision. Besides, while all other variables are fixed, the auditing threshold $\hat{\theta}$ declines as price increases. The broker can charge for higher price by providing stable service as advertised. However, this requires that the computing resources will always be available at any time. Thus regular audit is required for broker to ensure. On the other hand, compensation *b* for not complying with QoS will be made, this leads to the profit loss for broker.

Audit is required for service broker to prevent from profit loss. High sunk cost will be inevitable if the provider deviates frequently, and this leads to high processing cost. The sunk cost can be decreased only when the providers always provide sufficient computing resources. In order to achieve this, the broker will choose to audit.

PROPOSITION 3 (Audit Strategy) The audit decision is (i) positively associated with opportunity cost of the service provider, (ii) but negatively with the contract reward, contract penalty, the price to the service provider. That is $\partial \alpha^* / y_1 > 0, where \ y_1 \in \{v\}$ *and* $\partial \alpha^* / y_2 < 0, where \ y_2 \in \{s, t, w\}$.

The audit threshold increases as the reward increases. This observation is quite interesting. From the perspective of broker, it is expected that providers will always supply sufficient computing resources if high incentive is given. This means that broker has belief that the incentive mechanism is helpful and audit is not necessary. However, penalty is also helpful to control the probability of audit. High penalty will decrease the audit threshold, thus low down the need of audit. The broker may decide to adopt high penalty policy if he thinks the provider is unreliable. On the other side, high penalty may frighten the broker psychologically and deviation will be diminished.

From the business view, which one is best choice to low down the audit probability? The main purpose of utility computing is to provide users with computing resources which are easily accessible and switchable. Users can obtain services from other brokers without paying high switching cost. If high price is charged, loss of customer base may be inevitable. Lowing down sunk cost can also help, but this is partially dependent on the behavior of service provider. Sunk cost can be lower only when the broker has confidence that provider seldom deviates. However, the uncertainty is unpredictable, thus making investment decision under this situation may be too risky. High compensation may be good for users in some ways, but is not desirable for broker.

The service provider will always choose to deviate if the opportunity cost is greater than the threshold $(s+t+w)/2$. This finding is quite straightforward. Because the reward, penalty and service fee are controllable by service broker, he can manipulate them carefully and avoid exceeding the opportunity cost of service provider. For the service broker, in order to maintain the QoS agreement, the service broker can either pay high service fee, or provides high reward to the provider. For service broker, extra effort is needed to audit, and doing this too frequently may not be feasible. From the view of broker, the most important part is how to balance between these factors. Now we discuss the situations under fixed opportunity cost. Service fee is paid by per-usage basis, but reward and penalty don't. If the broker has confidence that provider seldom deviates, the best strategy is providing low service fee with high reward and penalty.

3.5. Impacts of Reward and Penalty on Strategy Adoption

As observed from equation (1) and (2), the best response strategy is influenced by reward and penalty. The audit threshold increases as the reward increases, meanwhile deviation threshold decreases. This finding has an important managerial implication. From the view of service broker, no audit is required under high reward. On the other hand, this also reduces the probability of deviation, thus a mutual trust mechanism can be established by providing high reward. In contrast to reward, penalty may play a totally different role. High penalty frightens the provider and result in low deviation. However, the broker doesn't benefit from high penalty. Our analysis shows totally different result. Low audit is not shown as was expected; this conclusion and the intuition are opposite. The provider may deviate because they will be heavily punished for just a small portion of not complying with QoS.

PROPOSITION 4. (i)Reward has decisive influence on the auditing and provisioning strategies. (ii) For service broker adopting reward as incentive is better off than penalty.

Figure 5-8 are plotted with parameters *b*=100, *c*=150, *p*=200, *w*=100, *v*=100, *s* from

20 to 300 while *t* fixed to 100, *t* from 200 to 3000 while *x* fixed to 100. As observed from figure 7, the provider always makes profit under increased reward condition, and the broker also gets positive profit while $s \leq 260$. However, figure 8 indicates that the profit of provider decreases as penalty increases, and even though converged to 0, the broker makes no positive profit. Compared these two figures, we can conclude that both broker and provider will choose reward as incentive, and this helps to verify proposition 4.
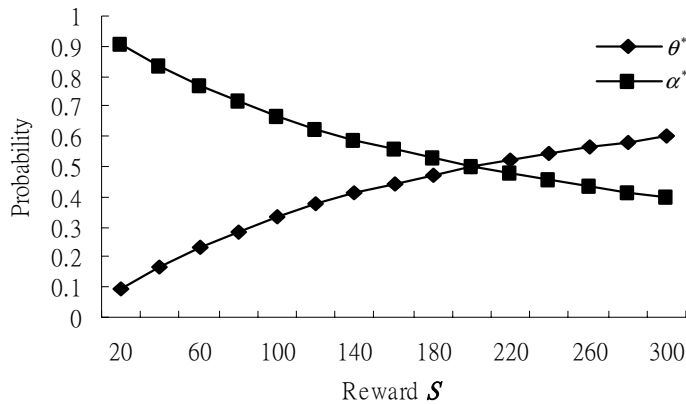

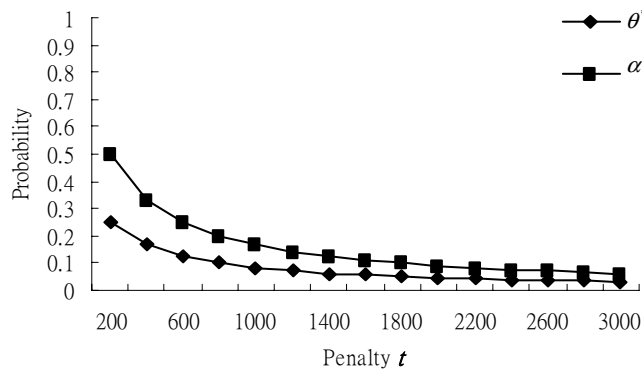Figure 5: Fixed Penalty with Increased Reward

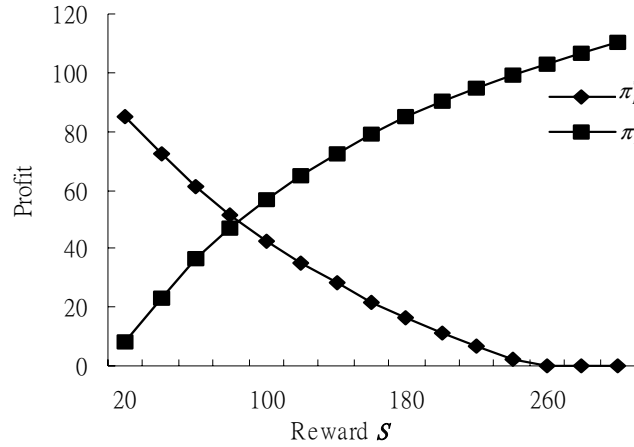Figure 6: Fixed Reward with Increased Penalty

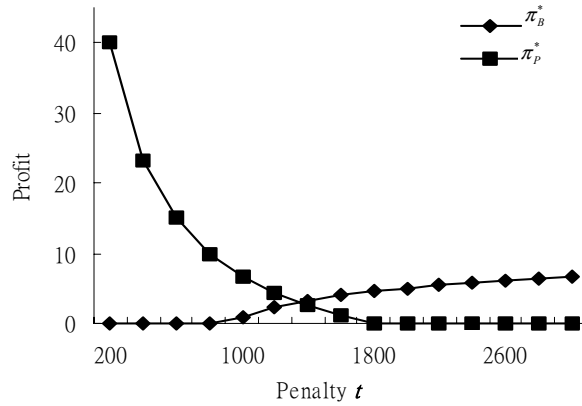Figure 7: Expected Equilibrium Profit against Increased Reward

Figure 8: Expected Equilibrium Profit against Increased Penalty

3.6. Pure Strategy in Nash Equilibrium

After discussing the mixed strategies of broker and provider, we turn our focus to the condition of pure strategy Nash equilibrium. If the Nash equilibrium falls into (Audit, Violate), that is, the broker audits and the provider violates, these two conditions must be satisfied: $(p+t)-c > -b$ and $v-(t+k) > w+s-(k+v)$. These imply that both conditions must be satisfied: $c < (p+t+b)$ and $v > (w+s+t)/2$. Once the broker audits and the provider does not violate (i.e., (Audit, Not Violate)), then $p-(w+s) > p-w$ and $w+s-(k+v) > v-(t+k)$ have to be held, that is, $s < 0$ and $v < (w+s+t)/2$. Follow the same procedure and we can easily conclude that while the broker does not audit and the

provider violates, $c > (p+t+b)$ and $v > 0$ are derived. If both does not follow the contract, then $p-w > p-(w+s)$ and $w-(k+v) > (w+v)-k$, and thus $s > 0$ and $v < 0$.

The observation that the extreme situations of cooperative and non-cooperative results of the utility computing supply chain from the existence of pure Nash equilibrium can be illustrated as figure 9.
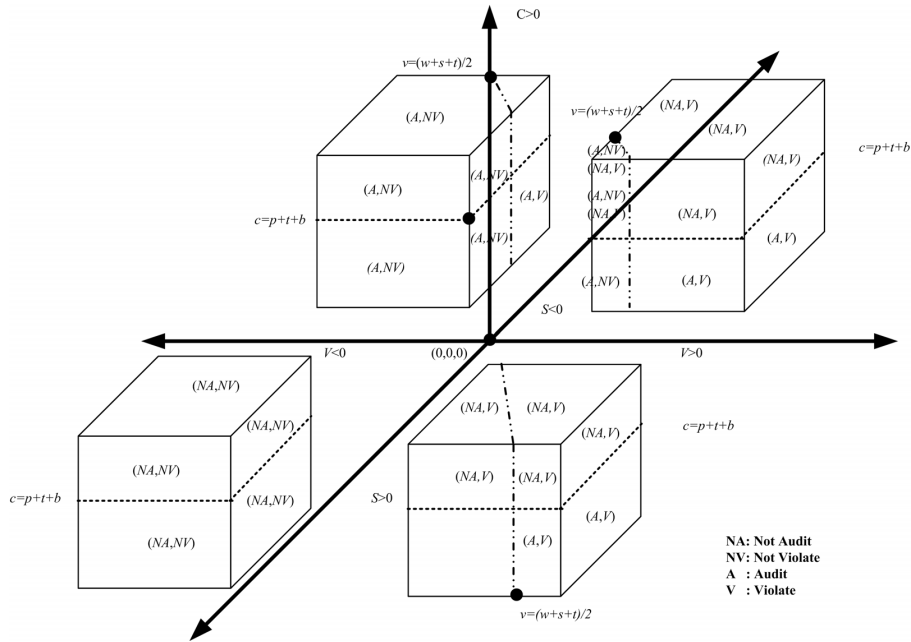

Figure 9: Pure Strategy Nash Equilibrium

PROPOSITION 5 (Pure Nash equilibrium)

$$\text{Strategy of } (\text{Broker,Provider}) = \begin{cases} (\text{NA, NV}) & \text{if } s > 0 \text{ and } v < 0 \\ (\text{NA, V}) & \text{if } c > (p+t+b) \text{ and } v > 0 \\ (\text{A, NV}) & \text{if } s < 0 \text{ and } v < (w+s+t)/2 \\ (\text{A, V}) & \text{if } c < (p+t+b) \text{ and } v > (w+s+t)/2 \end{cases}$$

Where A: Audit, V: Violate, NA: Not Audit, NV: Not Violate

Provider gets no profit from violating QoS if there exists no opportunity cost and reward is provided. Thus broker won't audit and provider always complies. Even though opportunity cost does exist, if it is not high enough ($v < (w+s+t)/2$), provider is still willing to comply. However, broker may worry about provider's violation because of no reward. Thus audit is reasonable for broker.

The first interesting finding is that broker won't audit under high transition cost ($c > (p+t+b)$). As we stated, transition cost is proportional to sunk cost of service broker. The decision of how much to be invested is based on the estimation of service demand. Costs of idle facilities or idle capacity means costs such as maintenance, repair, housing, and other related costs. Extra idle cost has to be spent if broker over-estimates. Under this situation, the violation of provider helps to reduce idle cost and therefore audit is not required.

The second finding is quite counterintuitive. We may imagine that if the transition cost is relatively low ($c < (p+t+b)$), broker can easily provide computing resources to requestor even if provider chooses to violate, therefore audit is unnecessary. However, the provider will always deviate if the opportunity cost is sufficiently high ($v > (w+s+t)/2$), thus all service requests must be frequently redispatched. Under this situation, broker will face the capacity problem because of heavy requests, and high investment on facilities is inevitable. Besides, higher probability of paying compensation to service requestor will reduce total profit. In order to reduce investment and lower down compensation, the broker has to audit.

4. CONCLUSIONS

Utility computing facilitates strategic agility by making available computing services and business process components on an as-needed basis (Ross & Westerman 2003). It starts by asking what the user wants. Ask any business manager how they want to get their IT resources and the answer will be likely "reliably and cost-effectively." However, as the IT technology becomes more complex, it may be difficult for single service provider to offer all the IT resources required by enterprise. In this paper we introduce the service broker who acts as a resource allocator and coordinator into utility computing business, and discuss the competition strategy between service broker and service provider. Users always seek for reliable utility computing service, and the service broker have to provide QoS guarantee. The service broker then has to acquire computing resources from internal or external service providers. However, a single service provider can serve several service brokers simultaneously. In order to make sure that users' requirement can be fulfilled at all time, an incentive and audit mechanism is built.

In our paper best response functions under mixed strategy for service broker and provider is analyzed. Audit decision is fully controlled by broker himself, while the deviation probability is controlled by both broker and provider. Besides this, reward is more efficient than penalty to make service providers comply with QoS agreement. Overall

speaking, the service broker still dominates in utility computing service provisioning model. In mixed strategy, reward and penalty play important roles in make audit and deviate decision. Under pure strategy condition, in addition to reward, the computing cost of broker is taken into consideration in deciding whether to audit. Opportunity cost is equally important for both mixed and pure strategy in making deviation decision.

There are several directions for future research. In our model the price that users paid to service broker is assumed to be fixed and is not discussed. One possible extension is to take resources usage into consideration. The service charge is highly related to resources usage, and the reward and penalty should be positively related to this. Resource allocation is also an important issue in utility computing. Currently, the probability of dispatching service requests to specific service provider is not considered. It will be interesting to introduce the resource allocation mechanism into our model.

References

1. Aib, I., Salle, M., Bartolini, C., and Boulmakoul, A. "A Business Driven Management Framework for Utility Computing Environments," Trusted Systems Laboratory. HP Laboratories Bristol, 2004.

2. Aphrodite, T., and Thomi, P. "An Overview of Standards and Related Technology in Web Services," *Distributed and Parallel Databases* (12:2-3) 2002, pp. 135-162.

3. Bailey, J., and Bakos, Y. "An Exploratory Study of the Emerging Role of Electronic Intermediaries," *International Journal of Electronic Commerce* (1:3) 1997, pp. 7-20.

4. Bhargava, H.K., and Sundaresan, S. "Computing as Utility: Managing Availability, Commitment, and Pricing through Contingent Bid Auctions," *Journal of Management Information Systems* (21:2) 2004, pp. 201-227.

5. Buco, M.J., Chang, R.N., Luan, L.Z., Ward, C., Wolf, J.L., and Yu, P.S. "Utility Computing Sla Management Based Upon Business Objectives," *IBM Systems Journal* (43:1) 2004, pp. 159-178.

6. Chang, K., Dasari, A., Madduri, H., Mendoza, A., and Mims, J. "Design of an Enablement Process for on Demand Applications," *IBM Systems Journal* (43:1) 2004, pp. 190-203.

7. Chase, J.S., Anderson, D.C., Thakar, P.N., Vahdat, A.M., and Doyle, R.P. "Managing Energy and Server Resources in Hosting Centers," *ACM SIGOPS Operating Systems Review* (35:5) 2001, pp. 103-116.

8. Chen, P.-y., and Wu, S.-y. "New It Architecture: Implications and Market Structure," Workshop on Information Systems and Economics, Washington D.C, 2004.

9.   Chris, K., and Giorgos, C. "Grid Resource Commercialization: Economic Engineering and Delivery Scenarios," in: *Grid Resource Management: State of the Art and Future Trends*, Kluwer Academic Publishers, 2004, pp. 465-478.

10. Earl, M.J. "The Risks of It Outsourcing," *Sloan Management Review* (37:3) 1996, pp. 26–32.

11. Eichberger, J. *Game Theory for Economists*, Academic Press, San Diego, California, 1993.

12. Eilam, T., Appleby, K., Breh, J., Breiter, G., Daur, H., Fakhouri, S.A., Hunt, G.D.H., Lu, T., Miller, S.D., Mummert, L.B., Pershing, J.A., and Wagner, H. "Using a Utility Computing Framework to Develop Utility Systems," *IBM Systems Journal* (43:1) 2004, pp. 97-120.

13. Farrell, A.D.H., Sergot, M.J., Salle, M., Bartolini, C., Trastour, D., and Christodoulou, A. "Performance Monitoring of Service-Level Agreements for Utility Computing Using the Event Calculus," 1st IEEE Workshop on Econtracting, 2004, pp. 17-24.

14. Hellerstein, J.L., Katircioglu, K., and Surendra, M. "A Framework for Applying Inventory Control to Capacity Management for Utility Computin," Ixth IFIP/IEEE International Symposium on Integrated Network Management, 2005, p. 237.

15. Huang, K.-W., and Sundararajan, A. "Pricing Models for on-Demand Computing," Stern School of Business, New York University, 2005.

16. Huhns, M., and Singh, M.P. "Service-Oriented Computing: Key Concepts and Principles," *IEEE Internet Computing* (9:1) 2005, pp. 75-81.

17. Lodi, G., Panzieri, F., Rossi, D., and Turrini, E. "Sla-Driven Clustering of Qos-Aware Application Servers," *IEEE Transactions on Software Engineering* (33:3) 2007, pp. 186-197.

18. Machiraju, V., Bartolini, C., and Casati, F. "Technologies for Business-Driven It Management," Internet Systems and Storage Laboratory, HP Laboratories Palo Alto, 2004.

19. Machiraju, V., Rolia, J., and Moorsel, A.V. "Quality of Business Driven Service Composition and Utility Computing," Software Technology Laboratory, HP Labs Palo Alto, 2002.

20. Mani, A. "Oss-Based Grid Computing," 2007.

21. Monahan, B. "Infrastructure Security Modeling for Utility Computing," Trusted Systems Laboratory, HP Laboratories Bristol, 2005.

22. Paleologo, G. "Price-at-Risk: A Methodology for Pricing Utility Computing Services," *IBM Systems Journal* (43:1) 2004, pp. 20-31.

23. Pfister, G.F. *In Search of Clusters*, Prentice Hall, Upper Saddle River, 1998.

24. Rajeev, K., and Heungsoo, P. "A Cooperative Game Theory Model of Quantity Discounts," *Management Science* (35:6) 1989, pp. 693-707.

25. Rappa, M.A. "The Utility Business Model and the Future of Computing Services," *IBM Systems Journal* (43:1) 2004, pp. 32-42.

26. Ross, J.W., and Westerman, G. "Architecting New Outsourcing Solutions: The Promise of Utility Computing," Center for Information Systems Research, Massachusetts Institute of Technology, 2003.

27. Ross, J.W., and Westerman, G. "Preparing for Utility Computing: The Role of It Architecture and Relationship Management," *IBM Systems Journal* (43:1) 2004, pp. 5-19.

28. Rowley, J. "Principles of Price and Pricing Policy for the Information Marketplace," *Library Review* (46:3) 1997, pp. 179-189.

29. Sahai, A., Singhal, S., Joshi, R., and Machiraju, V. "Automated Policy-Based Resource Construction in Utility Computing Environments," Internet Systems and Storage Laboratory, HP Laboratories Palo Alto, 2003.

30. Sieber, P., and Griese, J. "Organizational Virtualness and Electronic Commerce," 2nd International VoNet-Workshop, Zurich, 1999, pp. 23-24.

31. Vorisek, J., and Feuerlicht, G. "Is It the Right Time for the Enterprise to Adopt Software-as-a-Service Model?," *Information Management* (17:3) 2004, pp. 18-21.

32. Werder, M.R. "Pricing in the Service-Oriented It World," in: *Department of Information Technology*, University of Zurich, Switzerland, 2004.

33. Yeo, C.S., and Buyya, R. "Service Level Agreement Based Allocation of Cluster Resources: Handling Penalty to Enhance Utility," 7th International Conference on Cluster Computing, Boston, MA, 2005.

34. Yeo, C.S., Buyya, R., Assuncao, M.D.d., Yu, J., Sulistio, A., Venugopal, S., and Placek, M. "Utility Computing on Global Grids," Grid Computing and Distributed Systems Laboratory, The University of Melbourne, Australia, 2006.