International Journal of Grid and Distributed Computing
Vol. 9, No. 9 (2016), pp.251-268
http://dx.doi.org/10.14257/ijgdc.2016.9.9.22

A Cloud Service Broker Based on Dynamic Game Theory for Bilateral SLA Negotiation in Cloud Environment

Hongwei Chen, Xiaojuan Liu, Hui Xu and Chunzhi Wang

School of Computer Science, Hubei University of Technology, Wuhan, China
chw2001@sina.com

Abstract

In recent years, with the development of cloud computing, more and more service providers appear, making the cloud computing market more open and competitive. However, it is difficult for service consumers to choose the most suitable service provider for their own, especially for the service providers to provide the best service in the negotiated price, while the service providers and the service consumers themselves can't automatically negotiate, thus in this paper, we propose a cloud service broker framework based on dynamic game theory for bilateral SLA negotiation in cloud environment, aiming for successful negotiation among participators. This paper also proposes a Nash equilibrium point and a satisfaction degree difference algorithm, aiming to find the optimal value of SLA attributes (price and bandwidth) at the minimum satisfaction degree difference among the participators. Experimental results show that the service consumers and the service providers are able to achieve the same satisfaction degree on price and bandwidth.

***Keywords**: Cloud Environment; Cloud Service Broker; SLA Negotiation; Dynamic Game*

1. Introduction

Recently, cloud computing has been a hot topic in the research area, and also an attractive technology in the field of computer science. By providing a new type of service for the users, cloud computing is changing our life. The users can obtain services from the cloud without paying attention to the details [1]. In cloud computing, the cost of the online service is calculated on the basis of usage, service consumers don't need to keep a long term contract with service providers. In this case, the service consumers can save a lot of expenditure budget on operation, management and transmission services. The cloud computing can be described as a new form of IT environment, which provides dynamic, flexible and scalable virtual resources. The definition of cloud computing has varieties of claims, but the widely accepted one is defined by American National Institute of standards and Technology (NIST): cloud computing is a pay by usage mode, which provides available, convenient, on-demand network access, entering the configurable computing resources (the shared pool resources including network, server, storage, application software, services). Those resources can be provided rapidly, only need a few inputs management or very little interaction with the service provider [2].

At present, the service consumers can use different cloud providers, which mainly include: Amazon Ec2, Azure from Microsoft, Google Application Engine, GOGrid, CloudSigma *etc.* These cloud providers perform different tasks on the instance of a particular type, and provide different instances to meet different service requests in specific price. In cloud computing, Service Level Agreement (SLA) is an important part of the service contract. And in most cases, it is negotiation protocol between service providers and service consumers. SLA should ensure both business success and customers' satisfaction degree. From the view of service provider, SLA helps to realize the goals of enterprise, which usually seeks to increase profit, and reduce the non-performance responsibility resulting from the unpredictable network or server interruption.

Similarly, from the view of service consumer, SLA tries to ensure acceptable service performance, and meet the needs of service consumers as much as possible [3]. Because resources are heterogeneous, and relying on the service consumer's individual needs and preferences, to select specific resources for task execution in lower prices, the negotiation process is complex and changeable. The negotiation process of SLA is necessary, because each service consumer is independent, and has different goals and bandwidth requirements. However, the SLA negotiation process should be automatic, because it is impossible for service providers and service consumers themselves to carry on the negotiation process, and achieve a common acceptable agreement [4]. This paper adopts the statement game of incomplete information dynamic game model, which focuses on how people transmit information through oral or written statement in the private information and information asymmetry situation. In order to balance the effectiveness and success rate, in this paper, it is necessary to design an automated bilateral SLA negotiation mechanism to satisfy the different needs of the service consumers with different preferences in an uncertain and dynamic service environment.

In this paper, the structure is as follows. The next section will introduce the related research work. The third part of this paper introduces the architecture of cloud service broker as broker between service providers and service consumers, and explains the different module. In the fourth part, this paper proposes a SLA negotiation model based on game theory. The fifth, sixth and seventh part respectively introduces the mathematical model, algorithm design and analysis of experiments. The eighth part is the summary of this paper and a list of future research.

2. Related Work

At present, the standardization work has conducted an extensive research on SLA in cloud computing. Among them, the cloud service broker and SLA management, especially the SLA negotiation is the subject of some research efforts in the past few years. And with the surge of cloud service, cloud service broker stand out and mainly focus on the utilization, performance and cloud service delivery management, and lead to a number of negotiations between service consumers and service providers. Below we introduce some major research work:

In a joint cooperation organization model framework, Pouyllau Hélia and Giovanna Carofiglio [5] proposed a synthetic technique for inter carrier SLA to respect users' QoS requirements and simultaneously maximize the long-term benefits of network operators. Gangadharan [6] analyzed the current cloud computing SLA on whether it meets the users protect their data concerns rights (if any) and business expectations, and whether they can build trust between cloud consumers and cloud providers. A reference implementation of successful implement SLA computing infrastructure is introduced in [7]. This paper discusses limitations of the current infrastructure, *i.e.* service products, and introduces the realization of SLA function. The infrastructure SLA and service management has been in development, and extended to Apache Tashi and low level monitoring system. A common open interface for heterogeneous infrastructure has been developed, which has helped create the recently released open cloud computing interface (OCCI). Silaghi [8] designed a common framework of the time constrained service level values strategic negotiation, and learning agent by extending Bias to cope with the negotiation session duration, and to reflect the use of their framework. They proved that computing grid opponent learning strategies are worth considering in the open competition, which is mainly on the optimal allocation of resources and the participants justice satisfaction degree. Another research's focus is to develop a selection strategy model, aiming to select the appropriate strategies from a set of different strategies to solve or avoid meeting conflict [9]. Selection strategy model is based on the use of artificial intelligence technology to analyze historical data, the current meeting arrangements, the participants' personal data and preference data et cetera.

Farhana [10] proposed a novel trust negotiation agent framework, which performs bilateral negotiation between service providers and service consumers, which based on SLA adaptability and intelligence of high level business needs. They define the mathematical model mapping the business level requirements to layer parameters of decision functions, and hide the complexity of system of the parties. Anithakumari [11] proposed a new SLA violation detection framework, which renegotiations for the establishment of the SLA in multiple SLA violation cases. SLA renegotiation contributes to limit resources excessive supply, thus optimizes the usage of the resource. The proposed architecture, as a whole, may maximize the business level goals for the cloud provider. Xu [12] used game theory stochastic learning scheme in the opportunistic spectrum access (OSA) system, in which the channel availability statistics and the number of the secondary users is a distributed channel selection problem of priori unknown survey. They worked on a game channel selection problem, which is proved to be the exact potential game. Another paper solves the negotiation problem of coordination bidding strategy between an agent and multiple agents (*i.e.* one-to-many negotiation), and discusses the difference interdependent factors [13]. When negotiation subject to the current opponents' behavior, they suggest using negotiation bidding strategy in resource distribution/redistribution. The contribution of another paper is to design multi-issues negotiation mechanism, in order to promote the price, time slot, quality of service negotiate synchronously, including the design of utility function of the quality of service, and price, time slot, and the problem of the level of service quality which are based on the adaptivity and similarity compromise proposal [14]. Because cloud computing business negotiation is a dynamic and adaptive process, based on multi-agents system self-organization, another paper puts forward a method for realizing the macro properties of the cloud computing market local interactive agents requirement [15]. The establishment of the novel three layers self-organization multi-agents mechanism supports cloud commercial parallel negotiation activities. Another paper uses multi-agents system and exchange information protocol based SLA, and proposed an efficient job scheduling strategy. Meanwhile, based on the optimal use of available resources, namely the negotiation time and grid resources, this paper proposes a new algorithm. It also used the previous negotiation session information, not only improves the possibility of successful negotiation, but also reduces the negotiation time [16]. Another paper presents a general SLA negotiation protocol and a specification language for virtual environments [17].

Table 1. Comparative Analysis between this Paper and Other References

	This paper	**Other references**
Algorithm	The proposed Nash equilibrium point satisfaction degree differential algorithm, sets up a game matrix using the service provider and the service consumer satisfaction degree, finds out the Nash equilibrium point, and then compares the degree of satisfaction degree difference, until satisfaction degree difference is minimized and tends to 0.	Calculate all degree of satisfaction degree differences, and then compare the size [4]; Comparison of the value of utility function [16]; Similarity calculation as well as the priority [17];
Satisfaction degree formula	This paper puts forward nonlinear model, transaction period model and exponential function model respectively to establish the satisfaction degree formula.	Satisfaction degree is linear with the change of bandwidth or price [4];

Negotiation frame	By adopting statement game method of incomplete information dynamic game theory, the game process is clearly reflected. Based on increasing cloud service broker, increase of the game module can better reflect the dynamic game between service provider and service consumer.	Negotiation between a service provider and a service consumer [3]; Negotiation between service provider and service consumer on the basis of cloud service broker[7]; The negotiation between multi-agents system [15]; Negotiation between the service provider agent and service consumer agent [16];
Properties of SLA	Price and bandwidth;	The price and quality of service [4]; Price, time slot and the quality of service [14]; Price and reaction time [16];
Game method	Bilateral games between service providers and service consumers based on the cloud service broker;	One-on-one negotiation between a service provider and a service consumer [3]; One to many negotiation between the service consumer agent and service provider agent [13];

3. The SLA Negotiation Architecture Based on Cloud Service Broker

3.1. The SLA Negotiation Architecture

In Figure 1, the main modules of the system structure are: the service consumers, the cloud service broker, the monitoring center and service providers. In the following part, these main modules will be detailed.

(1) Resource Requirement Request (RRR)

Request details of service consumers are stored in RRR. When service consumer requests resource, the task can be executed at a small cost. The request task service consumer input has to be calculated, such as the type of task, that is, the task is CPU intensive, memory intensive, disk intensive or network intensive, and SLA template. These requests are then sent to the cloud service broker module to be analyzed, examined and verified.

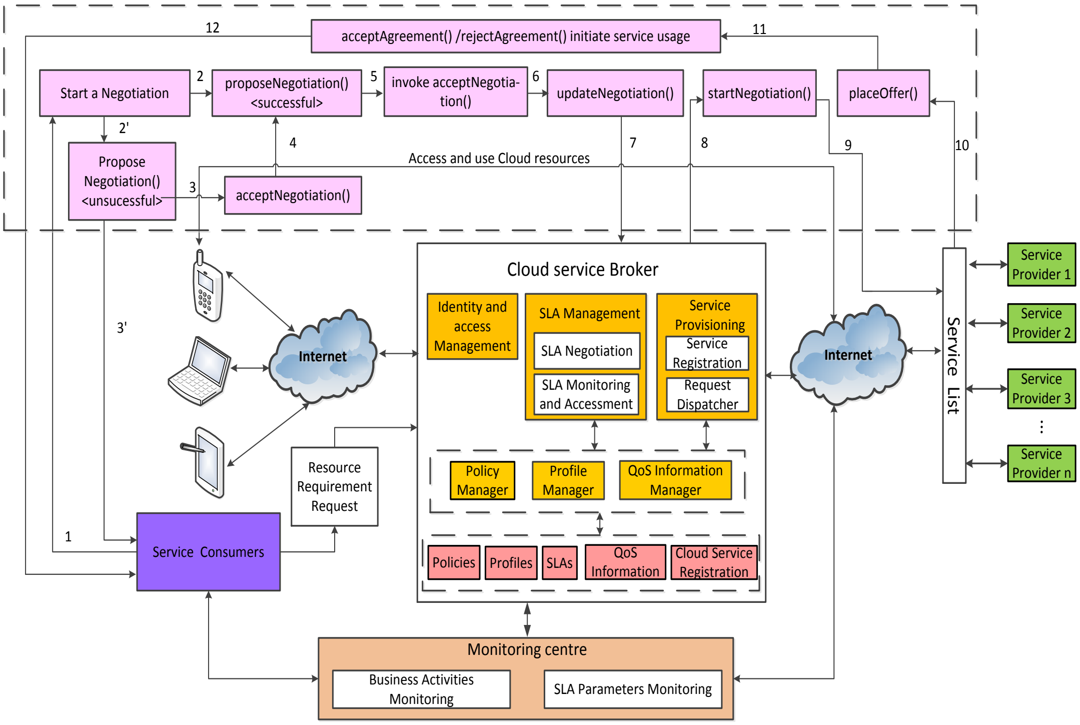

Figure 1. The SLA Negotiation Architecture Based on Cloud Service Broker

(2) Cloud service broker (CSB)

Considering that service consumers usually do not have the capability to negotiate, manage and monitor the quality of service, they appoint task to cloud service broker, such as choosing the appropriate service provider and SLA negotiation. As a third party mediation between service consumer and service provider, cloud service broker is responsible for processing the SLA requests submitted by service consumer, and sending these requests to the service list of the service provider, then conducting corresponding search and select the appropriate service, and negotiating with service provider.

Based on the service consumers' service demand and service providers' provision of service, request scheduler realizes management operation of service providers, and shall be responsible for selecting and implementing different strategies for appropriate service providers. The SLA manager achieves the SLA management operation, and is responsible for the SLA negotiation between the service consumer and the selected service provider. Then, the service consumer and service provider sign a contract. The contract describes the service type, ensured service level, service cost and punishment for preventing violating the agreement. If the selected service provider cannot achieve the desired level of service, cloud service broker will choose another service provider, and reiterate the negotiation process. The profile manager implements identity and access management operation, and is responsible for the management of the service consumer profiles, including the personalized service and the required service quality preference. Policy manager implements the policy management, and is responsible for the management of different policies, such as authorization strategy and service provider selection strategy of quality of service perception.

(3) The monitoring center

It is responsible for monitoring the service provider and the service consumer activities, quality of service and SLA protocol parameters *etc.*

(4) The service provider

Service providers can provide some service types that realize and use simple or composite services. In order to detect the existing services provided by the service provider, service providers need to use monitoring technology, allowing collecting the measurement data in the selected observation point. Through the summary of the collected data, service providers can detect index value of each service. If the current service provision has a significant decrease, service providers will add additional resources to meet SLA commitment. The SLA manager of service providers is responsible for the management of SLA template, negotiating the delivery service and quality of service level with cloud service broker or directly with service consumers, the duration of SLA, implementation of SLA and adjustment.

3.2. The SLA Negotiation Process

Figure 2 describes a SLA negotiation process scheme among service consumer, service broker components and selected potential service provider aiming to reach agreement for service provision. The negotiation process is divided into three steps: the expression of the service consumers' expectations, the selection of service provider, SLA negotiation. Each specific negotiation step as follows:

(1) Service consumers' service requests and SLA templates will be stored to Resource Requirement Request (RRR), which mainly stores service consumers' request details.

(2) Service consumer's requests are sent to cloud service broker, then cloud service broker finds an appropriate service provider for service consumer, satisfying its service functional and non-functional requirements.

(3) Analyzing and checking submitted requests from the service consumer, and after verifying consumer service, cloud service broker requests to configure file from the file manager. Then the cloud service broker requests to select an appropriate service provider from the selection manager. According to the service consumer demand, the service provider can deliver these services.

(4) Cloud service broker requests policy of selected service providers from the policy manager.

(5) Because service consumer previously used some services of cloud service broker, if the service consumer profile is available in the profile repository, cloud service broker can decide whether the selected service provider by the selection manager can handle the service consumer request. This decision depends on the service consumer and service provider configuration file.

(6) If the service consumer configuration file is not available in the profile repository. In order to create a new profile for the service consumers, the cloud service broker requests service consumers to provide some information, such as service preference and quality of service satisfaction degree et cetera.

(7) If at least one service provider can meet the needs of service consumer, the cloud service broker requests service provider to negotiate the terms of service and service delivery conditions from the SLA manager.

(8) The SLA manager of cloud service broker sends SLA requests to the SLA manager of service provider and requests a SLA recommendation, then the SLA manager of service provider parses the SLA request and verifies its SLA template.

(9) If the service provider accepts SLA requests, the SLA management of service provider sends back a SLA recommendation through cloud service broker to response SLA request. Cloud service broker analyses SLA recommendation, and determines whether it can satisfy all the functional and non-functional requirements of the service consumer.

(10) If SLA recommendation can meet all the needs of consumer services, cloud service broker accepts service provider's proposal, and sends a SLA confirmation to the SLA manager of service provider. Otherwise, the service broker refuses the offer and presents different conditions, terms, costs corresponding suggestions.

(11) In case of SLA confirmed, the service consumer and service provider of two parties pass agreement, and the service consumer can use service according to the terms. The agreement provides the service type to service consumer the service provider should provide, ensures the quality of service level, service cost, period of validity, and takes actions when the agreement is violated, and so on.

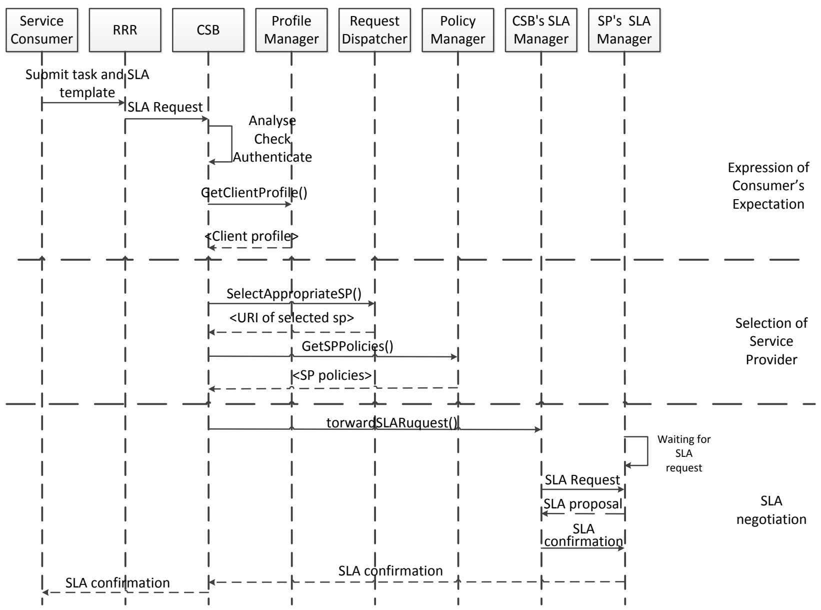

Figure 2. The SLA Negotiation Process

4. The SLA Negotiation Model Based on Dynamic Game

According to the design of the SLA negotiation process, proposes a nonlinear model, transaction period model and exponential function model, and then compares and analyzes the three models.

4.1. The Nonlinear Model

(1) Participants: This game has two players P=$(\mathrm{p}_1,\mathrm{p}_2)$, in which p_1 refers to the service consumer, p_2 refers to the service provider. For participant p_1, price is the most important parameter. For the participant p_2, bandwidth is the most important parameter. The two participants will try to seek the best strategy for the game.

(2) Set of strategies: Set of strategy profile defined by participants p_1 and p_2 are provided by $\mathrm{A}=\mathrm{S}_1\times\mathrm{S}_2$, where

$$\mathrm{S}_1=\{(\mathrm{P}_0),(\mathrm{P}_0+\Delta),(\mathrm{P}_0+2\Delta),(\mathrm{P}_0+3\Delta),(\mathrm{P}_0+4\Delta)\}$$

$$\mathrm{S}_2=\{(B_0),(B_0+\delta),(B_0+2\delta),(B_0+3\delta),(B_0+4\delta)\}$$

B_0 is offered bandwidth; P_0 is offered price; δ is the bandwidth parameter; Δ is the price parameter. The game matrix of strategy game for participants p_1 and p_2 can be given in Table 2.

Table 2. The Strategy Game Matrix

p_1 \ p_2	B_0	$B_0+\delta$	$B_0+2\delta$	$B_0+3\delta$	$B_0+4\delta$
P_0	0.5243,0.5212	0.5383,0.5081	0.5526,0.4947	0.5670,0.4812	0.5816,0.4675
$P_0+\Delta$	0.5142,0.5316	0.5282,0.5184	0.5424,0.5051	0.5569,0.4915	0.5715,0.4778
$P_0+2\Delta$	0.5039,0.5421	0.5180,0.5289	0.5322,0.5156	0.5466,0.5021	0.5612,0.4884
$P_0+3\Delta$	0.4935,0.5528	0.5076,0.5396	0.5218,0.5263	0.5362,0.5127	0.5508,0.4990
$P_0+4\Delta$	0.4830,0.5636	0.4970,0.5504	0.5113,0.5371	0.5257,0.5235	0.5403,0.5098

(3) Benefits: In this paper, the concept of benefits is used to measure satisfaction degree that the service consumer receives from the service provider for given the price and the bandwidth attributes, vice versa. In this paper, the utility function $U_c(\mathrm{P},\mathrm{B})$ or $U_c(\mathrm{P},\mathrm{B})$ in a two-dimensional space is curve and monotonous model. In the SLA negotiation, according to the application scenarios and different attributes characteristics, different participants usually have different utility function for each SLA attribute. Assuming price is the SLA attribute to be negotiated. From the service consumer's point of view, the lower the price is, the more the revenue is, and with a very high return value; however, from the perspective of service providers, the higher the price is, the more the revenue is, and with a very high return value. In this paper, we model utility function on price and bandwidth for service consumer and service provider.

Service consumer's utility function: If given price from the service provider is P_0, then service consumer's satisfaction degree on price $S_c(\mathrm{P}_0)$ can be obtained by the figure 3. According to the properties of similar triangles, satisfaction degree on price formula of service consumer can be deducted as follows:

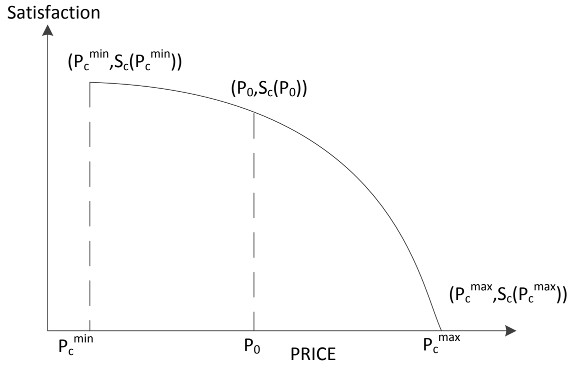

Figure 3. Satisfaction Degree of Price of Service Consumer

$$\frac{P_c^{\max^2}-P_0^2}{P_c^{\max^2}-P_c^{\min^2}}=\frac{S_c(\mathrm{P}_c^{\max})-S_c(\mathrm{P}_0)}{S_c(\mathrm{P}_c^{\max})-S_c(\mathrm{P}_c^{\min})} \tag{1}$$

$$S_c(\mathrm{P}_0)=\begin{cases} S_c(\mathrm{P}_c^{\max})-\dfrac{P_c^{\max^2}-P_0^2}{P_c^{\max^2}-P_c^{\min^2}}(S_c(\mathrm{P}_c^{\max})-S_c(\mathrm{P}_c^{\min})) & \mathrm{P}_c^{\min}\le \mathrm{P}_0\le \mathrm{P}_c^{\max} \\ 0 & \mathrm{P}_0>\mathrm{P}_c^{\max} \\ 1 & \mathrm{P}_0<\mathrm{P}_c^{\min} \end{cases} \tag{2}$$

However, service consumer's satisfaction degree on bandwidth increases gradually with bandwidth increasing. And if bandwidth is reduced, the satisfaction degree is also gradually reduced. Therefore, the satisfaction degree formula of service consumer is given below:

$$S_c(B_0)=\begin{cases} S_c(B_c^{\min})+\dfrac{B_0^2-B_c^{\min^2}}{B_c^{\max^2}-B_c^{\min^2}}(S_c(B_c^{\max})-S_c(B_c^{\min})) & B_c^{\min}\le B_0\le B_c^{\max} \\ 1 & B_0>B_c^{\max} \\ 0 & B_0<B_c^{\min} \end{cases} \tag{3}$$

The service provider's utility function: Similarly, for service provider, utility function on price and bandwidth can be derived as follows:

$$S_{sp}(P_0)=\begin{cases} S_{sp}(\mathrm{P}_{sp}^{\min})+\dfrac{P_0^2-P_{sp}^{\min^2}}{P_{sp}^{\max^2}-P_{sp}^{\min^2}}(S_{sp}(\mathrm{P}_{sp}^{\max})-S_{sp}(\mathrm{P}_{sp}^{\min})) & \mathrm{P}_{sp}^{\min}\le P_0\le \mathrm{P}_{sp}^{\max} \\ 1 & P_0>\mathrm{P}_{sp}^{\max} \\ 0 & P_0<\mathrm{P}_{sp}^{\min} \end{cases} \tag{4}$$

$$S_{sp}(B_0)=\begin{cases} S_{sp}(B_{sp}^{\max})-\dfrac{B_{sp}^{\max^2}-B_0^2}{B_{sp}^{\max^2}-B_{sp}^{\min^2}}(S_{sp}(B_{sp}^{\max})-S_{sp}(B_{sp}^{\min})) & B_{sp}^{\min}\le B_0\le B_{sp}^{\max} \\ 0 & B_0>B_{sp}^{\max} \\ 1 & B_0<B_{sp}^{\min} \end{cases} \tag{5}$$

The minimum satisfaction degree difference can be assumed as 0.001.

(4) Calculate the total utility function: Because the single SLA attribute utility function cannot evaluate suggested value of multiple SLA attributes. In order to draw general satisfaction degree for each service consumer and service provider on multiple attributes suggested values, the utility function of all the attributes need to be normalized into a common profit space, which is defined as the total income, by applying weight on each attributes income. The total income refers to overall satisfaction on multiple attributes proposals, and the weight value w refers to the factor of importance or preference of the corresponding attributes income $U(\mathrm{P},\mathrm{B})$ in total income space. In this research, the weight of price and bandwidth is set to the same value. Therefore, the total income on attribute price and bandwidth is shown as follows:

$$U_c(P_0,\mathrm{B}_0)=S_c(P_0)*\mathrm{w}_p+S_c(B_0)*\mathrm{w}_b \tag{6}$$

$$U_{sp}(P_0,\mathrm{B}_0)=S_{sp}(P_0)*\mathrm{w}_p+S_{sp}(B_0)*\mathrm{w}_b \tag{7}$$

Where $\mathrm{w}_p+\mathrm{w}_b=1$, the participants can change the weight value w_p and w_b according to their own preferences. These utility functions help to calculate the income value of each strategy played by participant p_1 and p_2. In this paper, for participant p_1 and p_2, two different absolute income values will be obtained.

4.2. The Trading Period Model

Assume that 24 hours of a day is divided into 4 time periods, and at each time period T has different values, that is T=[1,2,3,4]. Based on the proposed nonlinear model, the proposed transaction period model used to measure satisfaction degree of price and bandwidth for service consumer is shown as follows:

$$S_c(\mathrm{P}_0) = S_c(\mathrm{P}_c^{\max}) - \frac{P_c^{\max^T} - P_0^T}{P_c^{\max^T} - P_c^{\min^T}}(S_c(\mathrm{P}_c^{\max}) - S_c(\mathrm{P}_c^{\min})) \quad \mathrm{P}_c^{\min} \le \mathrm{P}_0 \le \mathrm{P}_c^{\max} \tag{8}$$

$$S_c(B_0) = S_c(B_c^{\min}) + \frac{B_0^T - B_c^{\min^T}}{B_c^{\max^T} - B_c^{\min^T}}(S_c(B_c^{\max}) - S_c(B_c^{\min})) \quad B_c^{\min} \le B_0 \le B_c^{\max} \tag{9}$$

Similarly, satisfaction degree of price and bandwidth for service provider can be drawn, such as formula (10) and (11).

$$S_{sp}(P_0) = S_{sp}(\mathrm{P}_{sp}^{\min}) + \frac{P_0^T - P_{sp}^{\min^T}}{P_{sp}^{\max^T} - P_{sp}^{\min^T}}(S_{sp}(\mathrm{P}_{sp}^{\max}) - S_{sp}(\mathrm{P}_{sp}^{\min})) \quad \mathrm{P}_c^{\min} \le \mathrm{P}_0 \le \mathrm{P}_c^{\max} \tag{10}$$

$$S_{sp}(B_0) = S_{sp}(B_{sp}^{\max}) - \frac{B_{sp}^{\max^T} - B_0^T}{B_{sp}^{\max^T} - B_{sp}^{\min^T}}(S_{sp}(B_{sp}^{\max}) - S_{sp}(B_{sp}^{\min})) \quad B_{sp}^{\min} \le B_0 \le B_{sp}^{\max} \tag{11}$$

The total utility function formula of service consumer and service provider is still the same as the formula (6) and (7).

4.3. The Exponential Function Model

Here, the exponential function model of the proposed single SLA attribute is used to measure satisfaction degree of the service consumer and service provider. For the-smaller-the-better attribute, the utility function is

$$U_1(x) = \mathrm{k}*(\mathrm{e}^{-ax} - \mathrm{b}) \tag{12}$$

Where U_1 refers to utility, x refers to the value of the attribute ($0 \le x \le 1$), a and b are constants, and k is a scale factor that calculate $U_1(0)=1$ and $U_1(1)=0$. Here, a can be assumed as 1, it follows that $\mathrm{k} = \frac{e}{e-1}$ and $b = \frac{1}{e}$ can be drawn. As a result, we get:

$$U_1(x) = \frac{e}{e-1}*(\mathrm{e}^{-x} - \frac{1}{e}) \tag{13}$$

Likewise, for the-bigger-the-better attribute, the utility function is

$$U_2(x) = \mathrm{k}*(\mathrm{e}^{ax} - \mathrm{b}) \tag{14}$$

Where U_2 refers to utility, x refers to the value of the attribute ($0 \le x \le 1$), a and b are constant, and k is a scale factor that calculate $U_2(0)=0$ and $U_2(1)=1$. Here, a can be assumed as 1, so $\mathrm{k} = \frac{1}{e-1}$ and $b=1$ can be drawn.

$$U_2(x) = \frac{1}{e-1}*(\mathrm{e}^{x} - 1) \tag{15}$$

So the total utility function of service consumer and service provider are:

$$U_c(\mathrm{p,b}) = U_1(\mathrm{p})*\mathrm{w}_p + U_2(b)*\mathrm{w}_b$$

$$U_c(\mathrm{p,b}) = \frac{e}{e-1}*(\mathrm{e}^{-p} - \frac{1}{e})*\mathrm{w}_p + \frac{1}{e-1}*(\mathrm{e}^{b} - 1)*\mathrm{w}_b \tag{16}$$

$$U_{sp}(\mathrm{p,b}) = U_1(b)*\mathrm{w}_b + U_2(p)*\mathrm{w}_p$$

$$U_{sp}(\mathrm{p,b}) = \frac{e}{e-1}*(\mathrm{e}^{-b} - \frac{1}{e})*\mathrm{w}_b + \frac{1}{e-1}*(\mathrm{e}^{p} - 1)*\mathrm{w}_p \tag{17}$$

where $\mathrm{w}_p + \mathrm{w}_b = 1, \quad 0 \le p \le 1, 0 \le b \le 1$, participants can change the weight value w_p and w_b according to their own preferences. In other references, they mentioned adjusting price model according to the remaining time. Obviously, in the process of negotiation, the remaining time of negotiation reduces gradually. Therefore, the less the remaining time is, the higher the price must be proposed by the user agent. So the negotiation time can be used as a parameter to determine the price value. With the reduction of remaining negotiation time, in the case of no agreements are reached, the agents always attempt to reach an agreement by setting price closer to their conservative prices. Therefore, for the user, the price of concession based on the remaining time is (18):

$$P(\mathrm{t}) = (1 - a(\mathrm{t})) * b(\mathrm{t}) + a(\mathrm{t}) * \mathrm{d}(\mathrm{t}) \tag{18}$$

Where

$$a(\mathrm{t}) = \mathrm{k} + (1 - \mathrm{k})(\frac{t}{t_{\max}})^{1/r} \tag{19}$$

$$d(\mathrm{t}) = p_c^{\max} - p_c(\mathrm{t} - 2) \tag{20}$$

$$b(\mathrm{t}) = Min[(1 - \frac{p_{sp}(\mathrm{t}-1)}{p_{sp}(\mathrm{t}-3)}) * \mathrm{p}_c(\mathrm{t}-2), \mathrm{d}(\mathrm{t})] \tag{21}$$

t is the number of current negotiation round, $t_{\max}$ denotes the maximum negotiation time, $a(\mathrm{t})$ is the time function, $p_c(\mathrm{t}-2)$ is the previous bid of the user agent, $p_{sp}(\mathrm{t}-1)$ and $p_{sp}(\mathrm{t}-3)$ are two previous bids of the provider agent, $b(\mathrm{t})$ denotes the simulation function, $d(\mathrm{t})$ is defined as a function which is considered to make the price close to the conservative price, the speed of negotiation is decided by γ, when $0 < \gamma < 1$, the agent can maintain the initial value until the time close to the negotiation deadline.

$$r = r_{\max} * (\frac{1 - sim(\mathrm{p}_c(\mathrm{t}-2), \mathrm{p}_{sp}(\mathrm{t}-1))}{t_{\max} - t}) \tag{22}$$

Where

$$sim(\alpha, \beta) = 1 - \frac{|\alpha - \beta|}{Max[\alpha, \beta]} \tag{23}$$

Formula (23) defines the similarity of price of the user and provider agents.

5. The SLA Negotiation Algorithm Based on Dynamic Game

Cloud service broker separately receives the maximum and minimum values of price and bandwidth from the service consumer and service provider respectively. Then the cloud service broker asks the service provider to select the initial values of price and bandwidth in the public range of price and bandwidth, denoted as (P_0, B_0), here we randomly select initial values in the public interval for simplification. According to the provided value, calculate degree of satisfaction of the service provider and the service consumer by utilizing degree of satisfaction model of service provider and service consumer. The values of price and bandwidth change from small to large to change with step length respectively, cloud service broker tries to find the best value for the service consumer and service provider. The game process is repeated for each round, in each round of game, computing pure Nash equilibrium value, and corresponding strategies are identified as the best strategy for the participants p_1 and p_2. Until reaching the Nash equilibrium point, satisfaction degree difference of between the service consumer and the service provider is close to zero, the game will end.

The algorithm in this paper is defined as the Nash equilibrium point and satisfaction degree difference algorithm, which does not need to calculate all variable values in game matrix, thus greatly reduce the calculation steps, being a reasonable method.

Table 3. The Pseudocode of the SLA Negotiation Algorithm

Algorithm 1. The SLA negotiation algorithm based on dynamic game

```
Begin
int round=1, max=4;
int $P_0$, int $B_0$;     // The SP randomly offers initial value in public interval
int $\Delta$, int $\delta$;     // step of the price P and bandwidth B respectively
struct S{
    double $S_c(P_{n+i\Delta})$; double $S_c(B_{n+i\delta})$;
    double $S_{sp}(P_{n+i\Delta})$; double $S_{sp}(B_{n+i\delta})$;
    double $U_c(P_{n+i\Delta}, B_{n+i\delta})$; double $U_{sp}(P_{n+i\Delta}, B_{n+i\delta})$;
}S;
 for(int n=0;n<=1;n++){
    for(int i=0; i<=max; i++){
        calculate $S.S_c(P_{n+i\Delta})$, $S.S_c(B_{n+i\delta})$, $S.S_{sp}(P_{n+i\Delta})$, $S.S_{sp}(B_{n+i\delta})$,
                  $S.U_c(P_{n+i\Delta}, B_{n+i\delta})$, $S.U_{sp}(P_{n+i\Delta}, B_{n+i\delta})$;
        GameMatrix=buildGameMatrix($S.U_c(P_{n+i\Delta}, B_{n+i\delta})$, $S.U_{sp}(P_{n+i\Delta}, B_{n+i\delta})$);
    }
    Equilibrium(n)=Nashequilibrium(GameMatrix) ;// find the Nash equilibrium point
    calculate $U(n)$ ;// Calculate satisfaction degree difference in Nash equilibrium point
 }
 $U_c = U(0); \ U_n = U(1);$
 while($|U_c| - |U_n| \geq 0$){
    for(int n=1;;n++){
        for ( int i=0; i<=max; i++){
            $(P_{c+i\Delta}, B_{c+i\delta}) = (P_{n+i\Delta}, B_{n+i\delta})$;
            Equilibrium(0)= Equilibrium(n); $U_c = U_n$;
            $(P_{1+n+i\Delta}, B_{1+n+i\delta}) = update(P_{c+i\Delta}, B_{c+i\delta})$;
            calculate $S.S_c(P_{1+n+i\Delta})$, $S.S_c(B_{1+n+i\delta})$, $S.S_{sp}(P_{1+n+i\Delta})$, $S.S_{sp}(B_{1+n+i\delta})$,
                      $S.U_c(P_{1+n+i\Delta}, B_{n+i\delta})$, $S.U_{sp}(P_{1+n+i\Delta}, B_{1+n+i\delta})$;
            GameMatrix=buildGameMatri ($S.U_c(P_{1+n+i\Delta}, B_{1+n+i\delta})$, $S.U_{sp}(P_{1+n+i\Delta}, B_{1+n+i\delta})$);
        }
        Equilibrium(n+1)=Nashequilibrium(GameMatrix) ;
        calculate $U(n+1)$; $U_n = U(n+1);$
    }
    round=round+1;
 } // Until the minimum satisfaction degree difference tends to 0
 wend
 return($P_0, B_0$)
 End
```

6. Simulation Results and Analysis

From the following three figures 4, 5 and 6, it can be seen price and bandwidths in the different ranges of the nonlinear model have different optimal values.

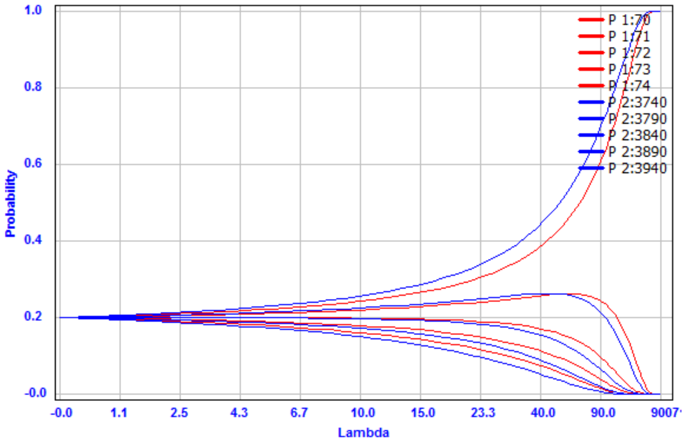

Figure 4. The Game Simulation Results Between the Service Consumer and the Service Provider

(at $P_0 = 70, \mathrm{B}_0 = 3740$ the minimum satisfaction degree difference reaches to 0.0030)

(1) The price range of service consumer and service provider: $P_c^{\max} = 100,\ P_c^{\min} = 55,$ $P_{sp}^{\max} = 105, P_{sp}^{\min} = 65$;The bandwidth range of service consumer and service provider : $B_c^{\max} = 4810, B_c^{\min} = 3120,\ B_{sp}^{\max} = 5240,\ B_{sp}^{\min} = 3630$;The initial values for service provider to provide: $P_0 = 68, \mathrm{B}_0 = 3640$,where P's step length is $\Delta = 1$,B's step length is $\delta = 50$; at $P_0 = 70, \mathrm{B}_0 = 3740$ the corresponding satisfaction degree difference reaches to 0.0030, which is the minimum satisfaction degree difference in interval of price and bandwidth provided by the service consumer and service provider, so the corresponding price and bandwidth values become the optimal value of the game.

(2) The price range of service consumer and service provider : $P_c^{\max} = 80, P_c^{\min} = 45,$ $P_{sp}^{\max} = 95, P_{sp}^{\min} = 60$ .The bandwidth range of service consumer and service provider : $B_c^{\max} = 4740, B_c^{\min} = 3330, B_{sp}^{\max} = 5730, B_{sp}^{\min} = 3630;$ The initial values for service provider to provide: $P_0 = 62, \mathrm{B}_0 = 3830$,where P's step is $\Delta = 1$,B's step is $\delta = 50$; at $P_0 = 75, \mathrm{B}_0 = 4480;$ $P_0 = 76, \mathrm{B}_0 = 4530;$ $P_0 = 77, \mathrm{B}_0 = 4580;$ $P_0 = 78,\ \mathrm{B}_0 = 4630;$ The corresponding satisfaction degree difference are all -0.0280,which is the minimum satisfaction degree difference in interval of price and bandwidth provided by the service consumer and service provider.

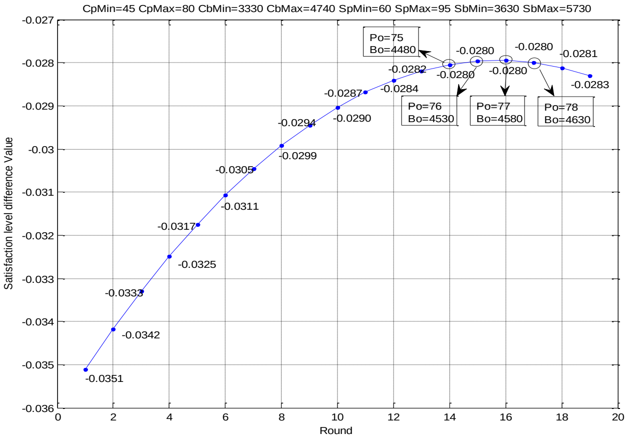

Figure 5. The Game Simulation Results Between the Service Consumer And the Service Provider

(at $P_0 = 75, \mathrm{B}_0 = 4480;\ P_0 = 76, \mathrm{B}_0 = 4530;\ P_0 = 77, \mathrm{B}_0 = 4580;\ P_0 = 78,\ \mathrm{B}_0 = 4630;$
The minimum satisfaction degree difference are all -0.0280)

(3) The price range of service consumer and service provider : $P_c^{\max} = 80, P_c^{\min} = 45,$ $P_{sp}^{\max} = 86, P_{sp}^{\min} = 50$ The bandwidth range of service consumer and service provider : $B_c^{\max} = 4920, B_c^{\min} = 3405,\ B_{sp}^{\max} = 5710, B_{sp}^{\min} = 3475$;The initial values for service provider to provide: $P_0 = 52, \mathrm{B}_0 = 3746$,where P's step is $\Delta = 1$,B's step is $\delta = 50$; at $P_0 = 53,\ \mathrm{B}_0 = 3796$ Degree of satisfaction degree difference reaches to 0.0008, which is the minimum satisfaction degree difference in interval of price and bandwidth provided by the service consumer and service provider. The example shows that the service consumer and service provider are able to achieve the same satisfaction degree in the price and the bandwidth, so the service consumer and service provider satisfaction degree difference can be close to an ideal situation.

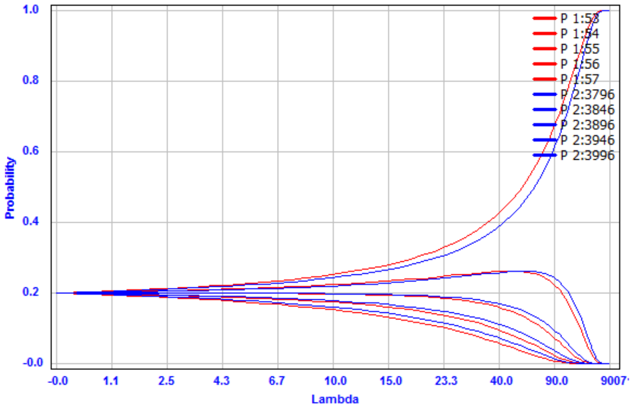

Figure 6. The Game Simulation Results Between the Service Consumer and the Service Provider

(at $P_0 = 53,\ \mathrm{B}_0 = 3796$ the minimum satisfaction degree difference reaches to 0.0008)

(4) Figure 7 is a comparison figure of satisfaction degree difference of three models. For a better comparison, the satisfaction degree difference is taken to be the absolute value. From comparison figure of three models, we can see that, in the same model, for the different ranges of values, the obtained satisfaction degree difference is different; however, in different models, for the same data range, the values of the corresponding

satisfaction degree difference are also different; but in every model, the corresponding satisfaction degree differences are all relatively small, and the results are also reasonable, the values of each data are as follows:

The first set of data:

$P_c = [55,100], B_c = [3120,4810], P_s = [65,105], B_s = [3630,5240], T = 1, P_0 = 68, B_0 = 3640;$

The second set of data:

$P_c = [45,80], B_c = [3330,4740], P_s = [60,95], B_s = [3630,5730], T = 2, P_0 = 62, B_0 = 3830;$

The third set of data:

$P_c = [60,85], B_c = [3120,5170], P_s = [56,87], B_s = [3650,5575], T = 3, P_0 = 63, B_0 = 3780;$

The fourth set of data:

$P_c = [45,80], B_c = [3405,4920], P_s = [50,86], B_s = [3475,5710], T = 4, P_0 = 52, B_0 = 3740;$

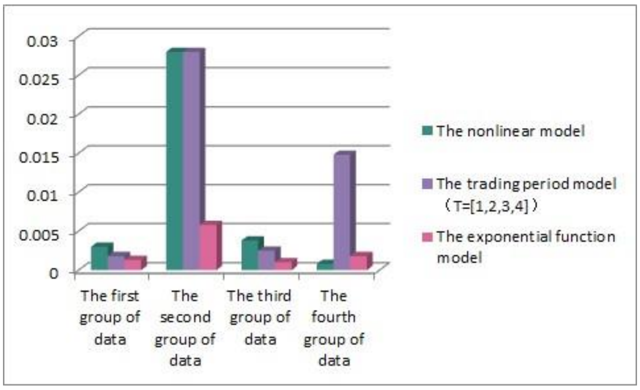

Figure 7. Comparison Chart of Satisfaction Degree Difference among Three Models

7. Conclusions

In the proposed framework, the negotiation between service providers and a service consumer is carried out based on the cloud service broker, and we also add some game modules in the original architecture, which can better reflect the dynamic game between a service provider and a service consumer. According to the negotiation process, we also introduce three models, namely: nonlinear model, transaction period model and exponential function model. The simulation results show that: for each model in different groups of data, the obtained satisfaction degree difference is different, and in the same set of data, different models calculate the satisfaction degree difference is also different; however, the precision of satisfaction degree difference can reach the ideal state.

In the negotiation about SLA parameters such as price and bandwidth, cloud service broker helps service consumers to find the most appropriate service provider. In this paper, we propose an infrastructure of cloud service broker based on dynamic game for bilateral SLA negotiation in cloud environment. However, this paper only considers the price and bandwidth as SLA properties. Later, in the negotiation process, we should try to consider multiple SLA attributes (such as response time, quality et cetera) or use the evolutionary game theory method for modeling negotiation, and maximize satisfaction degree of the service consumer and service provider.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 61170135, No. 61202287, No. 61440024), and the General Program for Natural Science Foundation of Hubei Province in China (No. 2013CFB020, No. 2013CFA046, No. 2014CFB484).

References

[1] Xu Gaochao, Junjie Pang and Xiaodong Fu, "A load balancing model based on cloud partitioning for the public cloud", TSINGHUA SCIENCE AND TECHNOLOGY, vol. 18, no. 1, (2013), pp. 34-39.

[2] Apostu Anca, Emanuil Rednic and Florina Puican, "Modeling Cloud Architecture in Banking Systems", Procedia Economics and Finance, no. 3, (2012), pp. 543-548.

[3] Zheng Xianrong, Martin Patrick, Powley Wendy and Brohman Kathryn, "Applying bargaining game theory to web services negotiation", 2010 IEEE International Conference on Services Computing (SCC), IEEE, (2010), pp. 218-225.

[4] Ray Benay Kumar, Sunirmal Khatua and Sarbani Roy, "Negotiation based service brokering using game theory", 2014 Conference on Applications and Innovations in Mobile Computing (AIMoC), IEEE, (2014), pp. 1-8.

[5] Pouyllau Hélia and Giovanna Carofiglio, "Inter-carrier SLA negotiation using Q-learning", Telecommunication Systems, vol. 52, no. 2, (2013), pp. 611-622.

[6] Gangadharan G. R. and Parrilli Davide Maria, "Service Level Agreements in Cloud Computing: Perspectives of Private Consumers and Small-to-Medium Enterprises", Cloud Computing for Enterprise Architectures. Springer London, (2011), pp. 207-225.

[7] John Kennedy, Andrew Edmonds, Victor Bayon, Pat Cheevers, Kuan Lu, Miha Stopar and Damjan Murn, "SLA-Enabled Infrastructure Management", Service Level Agreements for Cloud Computing, Springer New York, (2011), pp. 271-287.

[8] Silaghi Gheorghe Cosmin, Liviu Dan Şerban and Cristian Marius Litan, "A time-constrained SLA negotiation strategy in competitive computational grids", Future Generation Computer Systems, vol. 28, no. 8, (2012), pp. 1303-1315.

[9] S. M. Mozammal Hossain, "Selecting Negotiation Strategies for Meeting Scheduling Using a Model Based Approach", Procedia Computer Science, no. 10, (2012), pp. 1217-1220.

[10] Farhana H. Zulkernine and Patrick Martin, "An adaptive and intelligent SLA negotiation system for web services", IEEE Transactions on Services Computing, vol.4 , no. 1, (2011), pp. 31-43.

[11] S. Anithakumari and K. Chandra Sekaran, "Autonomic SLA Management in Cloud Computing Services", Recent Trends in Computer Networks and Distributed Systems Security, Springer Berlin Heidelberg, (2014), pp. 151-159.

[12] Xu Yuhua, Wang Jinlong, Wu Qihui, Anpalagan Alagan and Yao Yu-Dong, "Opportunistic spectrum access in unknown dynamic environment: a game-theoretic stochastic learning solution", IEEE Transactions on Wireless Communications, vol. 11, no. 4, (2012), pp. 1380-1391.

[13] Mansour Khalid, Ryszard Kowalczyk and Michal Wosko, "Aspects of Coordinating the Bidding Strategy in Concurrent One-to-Many Negotiation", Knowledge Engineering and Management, Springer Berlin Heidelberg, (2014), pp. 103-115.

[14] Son, Seokho and Kwang Mong Sim, "Adaptive and similarity-based tradeoff algorithms in a price-timeslot-QoS negotiation system to establish cloud SLAs", Information Systems Frontiers, (2013), pp. 1-25.

[15] Xu Jie and Jian Cao, "A Broker-Based Self-organizing Mechanism for Cloud-Market", Network and Parallel Computing, Springer Berlin Heidelberg, (2014), pp. 281-293.

[16] Arshad Sara and Nasrollah Moghadam, "A bilateral negotiation strategy for Grid scheduling", 2012 Sixth International Symposium on Telecommunications (IST), IEEE, (2012), pp. 592-597.

[17] Gomes Rafael L., Luiz F. Bittencourt and Edmundo R.M. Madeira, "A generic SLA negotiation protocol for virtualized environments", 2012 18th IEEE International Conference on Networks (ICON), IEEE, (2012), pp. 7-12.

Authors

Hongwei Chen, In 2006, he graduated from Nanjing University of Posts & Telecommunications and received PHD degree in China, majored in Communication and Information System. He is a professor at School of Computer Science in Hubei University of Technology, Wuhan, China. From August of 2013 to February of 2014, he was an academic visiting scholar at Temple University in USA. Now his major study field is Cloud Computing, Peer-to-Peer Computing and SDN.

Xiaojuan Liu, She is from Anhui Province of China, and a master candidate at School of Computer Science in Hubei University of Technology, interested in Cloud Computing and Game Theory.

Hui Xu, She received PHD degree in Radio Physics from Huazhong Normal University, Wuhan, China in 2010. Since 2006, she has been a certified computer system analyst in China. Now, she is an associate professor at the School of Computer Science in Hubei University of Technology, Wuhan, China. Currently, her major field of study is network and service management.

Chunzhi Wang, She is from Hubei province of China, PHD, professor and dean at School of Computer Science, in Hubei University of Technology. She is interested in Peer-to-Peer Computing and network security. She is a member of CCF, ACM and IEEE.