International Journal of u- and e- Service, Science and Technology
Vol.8, No. 7 (2015), pp.321-336
http://dx.doi.org/10.14257/ijunesst.2015.8.7.33

An SLA-oriented Multiparty Trust Negotiation Model based on HCPN in Cloud Environment

Chunzhi Wang, Qiuxia Chen, Hongwei Chen and Hui Xu*

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

Abstract

The negotiation on Service Level Agreement (SLA) in current complex cloud environment often involves the multilateral negotiation. Therefore, this paper presents an SLA-oriented multiparty trust negotiation model based on Hierarchical Colored Petri Net (HCPN) in cloud environment. This model mainly provides trust negotiation of SLA parameters such as credibility, reliability, availability and service price so on for Cloud Service Providers (CSPs) and Cloud Service Consumers (CSCs), thus making complex process description of multiparty trust negotiation on SLA in cloud environment simple and efficient. On the basis of the HCPN model on SLA, this paper puts forward a multi-objective optimization algorithm for SLA trust negotiation. Simulation results show that the algorithm not only improves efficiency of multilateral negotiation, but also avoids QoS negotiation falling into local optimization.

Keywords: *Multiparty Trust Negotiation, Hierarchical Colored Petri Net, Cloud Computing, Service Level Agreement, Genetic Algorithm*

1. Introduction

Service Level Agreement (SLA) is a signed agreement by consultation of Cloud Service Provider (CSP) and Cloud Service Consumer (CSC), which is not only to improve the safety and reliability of various access control of cloud services, and but also to be clear about the responsibility and obligation for the Cloud Quality of Service (QoS) [1]. The process of SLA negotiation involves trust negotiation mechanism of interacting parties, in order to solve problems of trust certificate disclosure and strategy in trust negotiation process.

Automated Trust Negotiation (ATN) is a kind of commonly used trust negotiation mechanism, and it is an access control method by gradually requesting and disclosing digital certificates between both sides of two strange entities to establish mutual trust, which is no longer dependent on third parties. But the cloud computing environment is open and complex, in which many parties often participate in the negotiation in the actual applications, and is no longer confined to only two sides of CSP and CSC to negotiate. The authentication in the cloud is no longer directly by mutual agreement, but by certification of authoritative department. There is a certain dependent relationship between the two entities in multilateral consultations. So in order to analyze the SLA trust negotiation, it can usually be decomposed into multiple ATNs, and each ATN develops authorization policies for each party to make them exchange authentication and trust certificates according to a certain order, which is a complicated process.

The Colored Petri Nets (CPN) [2] introduces the colors, in order to make a library expressed by class information and the complex structure greatly simplified. And it is also a very powerful tool of formally representing functions, which can better describe the process of MTN and concurrency, synchronization, conflict and sequential relationships for the system. Therefore, this paper adopts the tools of CPN to model and analyze the process of SLA negotiation, in order to make the SLA multilateral consultation process

ISSN: 2005-4246 IJUNESST

more intuitive and easy to understand.

The study of the SLA trust negotiation in cloud environment at home and abroad is mainly the trust negotiations from both sides. Oriol Farras, *et al.*, [3] proposes a method combining with the automated trust negotiation mechanism, which builds the trust relationship for strategies through the exchange of the digital certificates and trust certificates, so as to protect the privacy of both sides. Linlin Wu [4] puts forward the framework of automated trust negotiation for cloud computing, using an intermediary in negotiations between SaaS and multiple cloud service providers, in order to get different objectives, maximize profits and improve user satisfaction. Mohan Baruwal Chhetri, *et al.*, [5] puts forward a strategy-based SLA automatically specified framework for cloud computing services, making users and providers of cloud services to flexibly select the most appropriate scheme and supporting different interaction models.

In aspects of theory to design modeling with the use of CPNs, Christian Stahl, *et al.*, [6] introduces how to use CPN tools to design and analyze complex process, and create a model that is concise and easy to understand. Reference [7] provides a method with the use of CPN to formal modeling and analysis of the basic model of dynamic combination of Web services. As for large complex systems to establish a simple and intuitive hierarchical CPN model, hierarchical is a preferred choice. According to the characteristics of the hierarchical CPN model for easy dynamic simulation and verification, Reference [8] uses formal description and the modeling method of hierarchical CPN, transforms the Web service combination model described by the specific language into a hierarchical CPN model. Hierarchical CPN models can divide complex systems into several subnets, with each phase independently verified and incrementally refined, and it can perform dynamical simulation and formal verification of system behaviors, which makes up the lack of general Petri Nets [9].

The structure of this paper is organized as follows. Section 1 briefly introduces the SLA negotiation for cloud services and related research of CPN at home and abroad. Section 2 presents the framework of the MTN system in the cloud computing environment and functions of each component of the model. Section 3 introduces modeling theory and strategy language of hierarchical colored Petri net and it is attached to an SLA instance of multilateral negotiation. According to the system framework and policy language, Section 4 models and analyzes the instance of hierarchical colored Petri net. Section 5 is mainly based on the analysis of the model and puts forward a multi-objective optimization algorithm for the SLA negotiation and analyzes experimental results. Section 6 summarizes and puts forward the future work of the project.

2. An SLA-oriented Multiparty Trust Negotiation Architecture

Cloud computing environment places resources on the cloud side and makes a lot of resources for the user to choose, so often there are multiple CSPs can provide cloud services required by CSC. Before the use of cloud services, users need to perform SLA negotiation, and the result of SLA negotiation determines which CSP signs Cloud SLA Document (C_SLA Document). Trust negotiation of multiple parties contains the one-to-many model and the many-to-many model. This paper only studies the one-to-many model, and as for MTN, CSC no longer negotiates and specifies the CSP, but can negotiate with multiple CSP at the same time. This paper presents the architecture of the MTN trust model for SLA in cloud computing environment, as shown in Figure 1. Under the environment of multilateral negotiation, the SLA negotiation is mainly divided into two stages.

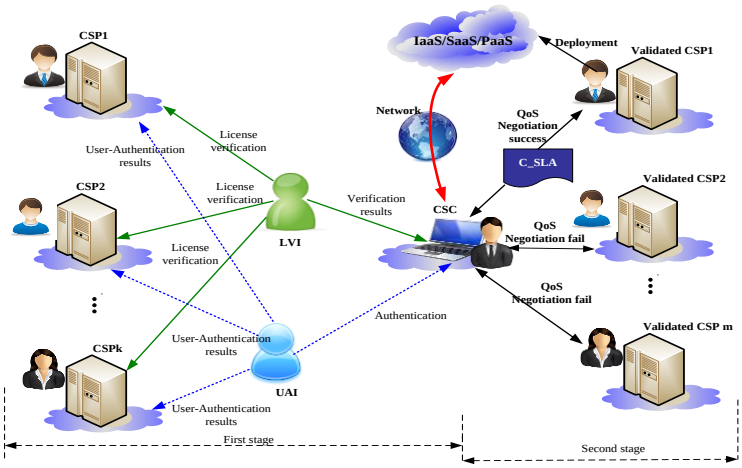

Figure 1. The Architecture of SLA-oriented Multiparty Trust Negotiation in Cloud Environment

The first stage of SLA negotiation is not negotiation of the CSC directly with the CSP parties, but it mainly consists of the four entities that are a CSC, multiple CSPs, Licence - Verification Institution (LVI) of CSP, User - Authentication Institution (UAI) for CSC users. Suppose that, LVI and UAI is trusted at the beginning, then after the CSP showing LVI trust certificate and CSC showing the trust certificate to UAI, LVI respectively verifies license to the CSP, conveys the verification results to CSC, and at the same time, UAI performs authentication to the CSC, and passes the certification result to the CSP respectively. After that, both trust sides go to the second stage.

The second stage focuses on the multilateral QoS negotiations, mainly including resources for cloud service, final negotiation CSP and CSC as SLA files (C_SLA), and multiple-negotiation entity.

(1) C_SLA files are files with legal effects produced by negotiation of the Cloud Service Provider (CSP) and CSC. According to the deployed resources of cloud services to CSC based on C_SLA file. The main components of its function are shown in Figure 2.

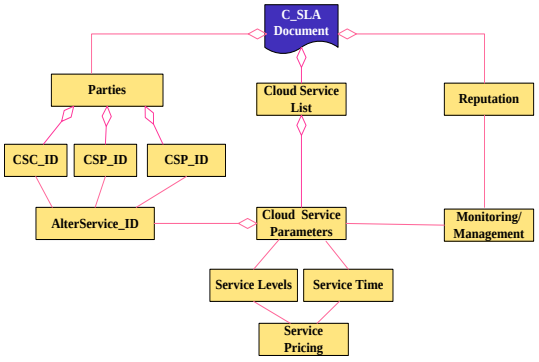

Figure 2. The Main Structure of SLA in Cloud Environment

(2) The cloud service list contains the service name, ID and service information, *etc.*, Parties participating in the SLA negotiation contain the ID information, optional serial number for cloud services (AlterService_ID). The selected Cloud Service ID corresponds to the corresponding cloud service parameters, which represent the service level, QoS, *etc.*, by measuring Service Pricing by computing the service level and the service time. In the process of service delivery, all kinds of information and service parameters for negotiation parties are monitored and manage, and the reputation is adjust and updated in a timely manner.

(3) QoS trust negotiation. The 6 attributes for QoS negotiation involved in this paper are Availability, Reliability, Reputation, Mean Time To Repair (MTTR), Time and Price. The Attributes of cloud services have different levels, expressed as $Q = \{QS_1, QS_1, ...QS_n\}$, $QS_k = \{Av_i, \mathrm{Re}_i, R, MTTR_i, T_i, \mathrm{Pr}ice\}$. After the service demand required by CSC, SLA negotiations are performed between CSC and each CSP. Finally, comprehensive QoS is evaluated, and in the MTN, the most optimal CSP is selected and the QoS negotiation for SLA ends.

(4) Resources for cloud services. The cloud computing technology has a strong ability of parallel processing for all computing resources, and provides the required resources in the form of service to users. According to the classification of service resources, cloud services are generally divided into three classes, which are Software as a Service, Platform as a Service and Infrastructures as a Service.

3. Access Control Policies on SLA-oriented Multiparty Trust Negotiation

A Multilateral Trust Negotiation (MTN) can be described by a policy Language to demonstrate dependent relationship between the negotiation parties and the demand. Based on this, the use of a systematic and automated way will divide multiple-party authorization into multiple-mutual negotiation and make them staggered according to a certain order. Therefore, the authorized model [10-12] is put forward, so that many trust negotiation can perform automated trust negotiation based on the authorization policy of each side. In this authorization mode, a Distributed Authorization and Release Control Language (DARCL) for MTN is put forward, and the authorization of DARCL can be decided according to the information content it receives or the information sender. So, DARCL is flexible as the policy language for MTN. The logic relationship of its strategy is described as Table 1.

Table 1. Policy Expressions and Descriptions

Policy Expressions	Policy Descriptions
$A \uparrow C \perp B$	A discloses trust certificate C to B
$A.trust(x) \leftarrow B.trust(x)$	The message that B trusts x is disclosed to A, then A will receives this information
$A \uparrow B.trust(E) \perp D \leftarrow B.trust(E)$	A receives the message that B trusts E, then A will tell D the message

In this paper, we study SLA multilateral negotiation of SaaS services in the cloud computing environment. This paper gives the following example. A cloud users (CSC) wants to rent an online management software (SaaS) from a cloud service provider (CSP), and if there are multiple cloud service providers at the same time providing the demand of the CSC, when making the SLA negotiation, it can be divided into two stages. The first stage is to identity authorization certification, and the second stage is the QoS negotiation phase. After the two stages of trust negotiation, the user can select a credible QoS-optimal CSP to sign SLA agreement.

In order to improve the reliability and security, only when the user provides effective authentication and authorization information to User Authentication Institution (UAI) for verification, CSP will provide the cloud services to CSC. In order to respect the privacy, only with the approval, UAI can provide the verification results of cloud user identity information to the CSP. The CSP discloses the trust certification signed by License Verification Institution (LVI) by the licensing authority. CSC verifies its authenticity and provides the trust certification required by the CSP. If at the same time two certified CSP

can provide the same type of service to the CSC, in QoS negotiation phase, CSC can perform SLA negotiation with two CSPs at the same time, then a better QoS can be obtained.

The first phase of the SLA negotiation party consists of three parts, which are CSC, CSPs and UAI. Usually a LVI is authorized to sign trust certification for CSP. Suppose that UAI and LVI trust each other, the relationship of identity authorization phase for the MTN negotiation is shown in Figure 3.

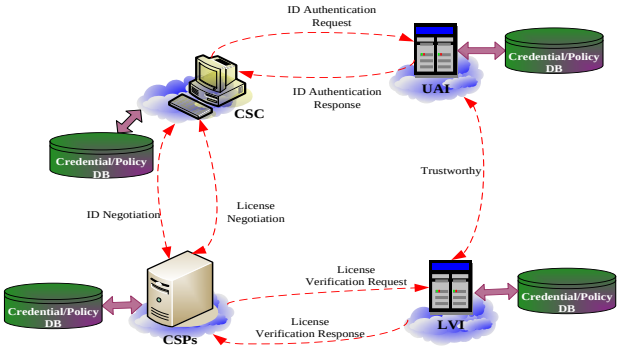

Figure 3. Relationship of Identity Authorization Phase for the MTN Negotiation

Negotiation policies for identity authorization for the SLA negotiation are described as follows.

(1) When CSP satisfies the following conditions, it will give the CSC online management software of cloud services. a) CSC discloses his identity information and authorization information and passes the CSP verification; b) CSP receives user authentication of CSC from the UAI.

(2) If the CSC discloses UAI authorization to the CSP, then the CSP will disclose it to UAI.

(3) Trust certification signed by the CSP and LVI can disclose to anyone.

(4) If the CSC passes the certification of UAI and allows UAI for providing the result to the CSP, the UAI discloses the trust certification of CSC for CSP.

(5) If the CSP is authorized as effective by the licensing authentication agency, CSC will authorize the trust certification of the identity information and disclose to CSP.

Description of the negotiation strategies of the parties in DARCL language is as shown in Figure 4.

Access Control Polices for the CSP:

$$CSP \uparrow CSP.service(x) \leftarrow x \uparrow Identity.information(x) \perp CSP \wedge x \uparrow x.release(UAI,CSP) \perp CSP \wedge UAI \uparrow UAI.clear(x) \perp CSP$$

$$CSP \uparrow x.release(UAI,CSP) \perp UAI \leftarrow x \uparrow x.release(UAI,CSP) \perp CSP$$

$$CSP \uparrow LVI.office(CSP) \perp x$$

Access Control Police for the UAI:

$$UAI \uparrow UAI.clear(x) \perp CSP \leftarrow CSP \uparrow x.release(UAI,CSP) \perp UAI \wedge UAI.clear(x)$$

Access Control Polices for the CSC:

$$CSC \uparrow CSC.release(UAI,x) \perp x \leftarrow x \uparrow LVI.office(x)$$

$$CSC \uparrow Identity.information(CSC) \perp x \leftarrow x \uparrow LVI.office(x) \perp CSC \wedge Identity.information(CSC)$$

Figure 4. Access Control Policies on SLA-oriented Multiparty Trust Negotiation

4. An SLA-oriented HCPN Model for Multiparty Trust Negotiation

4.1. The Top-level Diagram of the HCPN Model

The description of MTN consultation process for SLA is as follows. The ultimate goal of SLA multilateral negotiations is a certain sequence of negotiation between CSP, CSC, LVI, UAI, finally achieving their demand for the biggest satisfaction and generating a negotiation results in the negotiation between CSP and CSC, which is the SLA file, namely the CSC provides the appropriate price for the SLA negotiation of corresponding cloud services, and service providers also get needed profit and feedback. The SLA multilateral negotiations begin to first verify identity information, due to the negotiated authentication validations of multiple CSPs in the same way, so the identity authentication here does not consider the number of CSPs. Through the validation of QoS between each negotiation, then write the negotiation results into the cloud service list. FIFO queue processing is done to the QoS parameters of negotiation, according to the principle of First Come First Service to get detailed parameters and service levels for successful QoS negotiation from the cloud service list, Obtaining the parameters allow for many times to complete, and after getting the parameters and levels of the QoS, responsibility and guarantee of negotiations are done. Finally, an SLA document is produced, which completes the SLA negotiation at one time. Figure 5 shows the top-level diagram of CPN for the SLA multiparty negotiations.

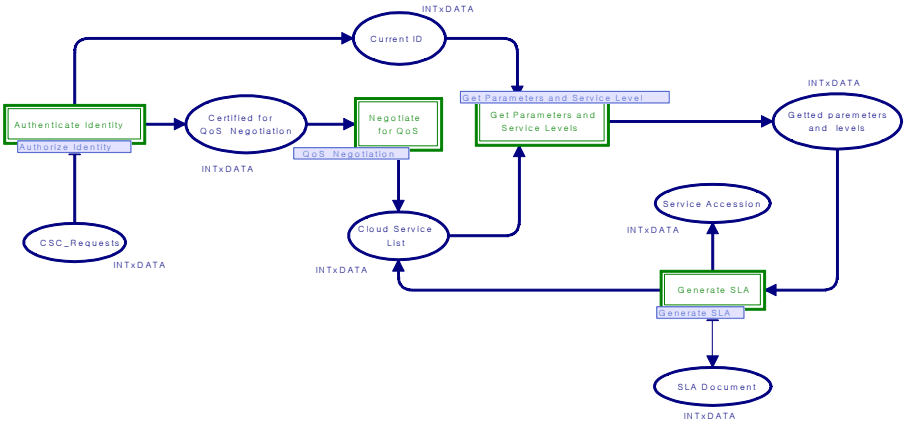

Figure 5. The Top-level Diagram of the HCPN Model

The top-level diagram uses four changing alternatives, among them, Authenticate Identity implements the identity authentication function of multiple negotiations for SLA negotiation; Negotiate for QoS implements the QoS of consultation; get parameters and service levels and obtain QoS parameters after QoS negotiation; generate SLA, achieve the successful consultation of SLA negotiation and generate effective agreements. These four alternatives change respectively in four sub-pages to complete the corresponding functions. Here only analyze the change of authenticate identity and negotiate for QoS, and the other two can be analyzed in the same way, and they are not discussed here.

4.2. Sub-diagrams of the HCPN Model

When the cloud user asks for identity authentication, cloud service providers will determine the authenticity and the legitimacy of the user according to the UAI certification and validation. If the users are true and legal, pass the authenticated users to the next stage in the QoS negotiation about SLA.

Figure 6 shows the CPN model of Authenticate Identity, which is established according

to the negotiation strategy shown in Figure 4. This model can describe the relationship between the cloud users and cloud service providers, as well as the relationship between the UAI and the LVI. According to the definition of global variables, ignore the same role, namely two CSPs have the same authentication, and in the sub-process only identity information of the different roles involved in negotiation are validated. CSC first initiates the authentication request, processes the request information, and after that, the trust certificate of identity information to the CSP. The premise condition $CSC\_Request$ is that. CSP passes the validation of LVI and presents a trust certificate, and trust certificate shown by CSP also depends on the results of authorized trust certificate $UAI.clear(x)$ from CSC inspection departments, in which x represents any entity. The library $LVI.office(CSP)$ is LVI official verification result of CSP, and the initial logo is set to three states, namely Credible, Incredible and Suspect. The library $x.release(UAI,CSP)$ is not restricted, meaning UAI trust certificate and CSP trust certificate released by x. $CSP\_Process$ means that CSP process has received information or has already known the information, according to the logic of negotiation strategy to build the dependencies between each others. If the parties agree, current certification ID and the next phase of the negotiation parties accredited by QoS negotiation are output.

Negotiation in this stage outputs credibility of cloud users and cloud service providers, which consists of an evaluation function to calculate. The evaluation function includes three aspects, which are the trust provided by the other party itself to support the certificate, the ratio of required certificates being satisfied in the interaction process, and whether the behavior of the other exchanging certificate accords with specification. The evaluation function is $F(t)=F_C+F_{pro}+F_{act}\cdot C_{need}$ means certificate sets needing each other to disclose, $g(C_i)$ means the set that certificate can meet. And the certificate set not disclosed is $C_{nod}=C_{need}-C_{dis}\cdot g(C_i)/|C_{need}|$ means the ratio of certificates being satisfied. If $|C_{need}|=q$, F_{act} means trust evaluation function of behavior. After the message m_k is sent from one party of the trust negotiation to the receiving party, if the message m_k is valid and trigger m_{k+1}, then $F(m)=1$, and vice versa.

$$F_{i,act}=\sum_i g(C_i)/q+\sum_i F(m_i)/|m| \tag{1}$$

The weight is introduced to set the parameters, and the evaluation function can be extended to

$$F(t)=\sum_i^{|m|}(\alpha_1 F(C_i)+\alpha_2 g(C_i)/|C_{need}|+\alpha_3 F_{i,act})=\sum_i^{|m|}\alpha_1 F(C_i)+\alpha_2 g(C_{dis})/|C_{need}|+\alpha_3 F_{i,act} \tag{2}$$

in which $\sum\alpha_i=1$, the function F_{C_i} means that certificate itself provides the trust degree, C_{dis} means the certificate sets disclosed by the other party, and the weight of disclosure strategy for trust certificate should be the biggest, so $\alpha_1>\alpha_2>\alpha_3$.

Figure 7 shows the CPN model of sub-page Negotiate for QoS, and in this stage, the CSC side negotiates the QoS respectively with two cloud service providers CSP1 and CSP2. This paper discusses six parameters of service properties for CSC to cloud service QoS requirements, including the feasibility, reliability, credibility, service execution time, mean time to failure repair and prices as $\{Av,Re,R,T,MTTR,P\}$. The library is $Certified\ for\ QoS\ Negotiation\in P_{In}$ and $Cloud\ Service\ List\in P_{Out}$, the color set INT × INT means state, types in the library are defined as INT × INT, and the library $CSC\_requirement$ contains the types and states of service attributes of the negotiation. $Policies(CSP1)$ and $Policies(CSP2)$ respectively means strategies received from the CSC

and strategies disclosed to the CSC, and after the change of $Release\ CSP1\_policies\_credentials$ and $Release\ CSP2\_policies\_credentials$, the trust negotiation strategy or trust certificate of CSP1 and CSP2 is passed to the CSC, and the negotiation results of the service attributes are passed to $Cloud\ Service\ List$, for the next negotiation of service properties.

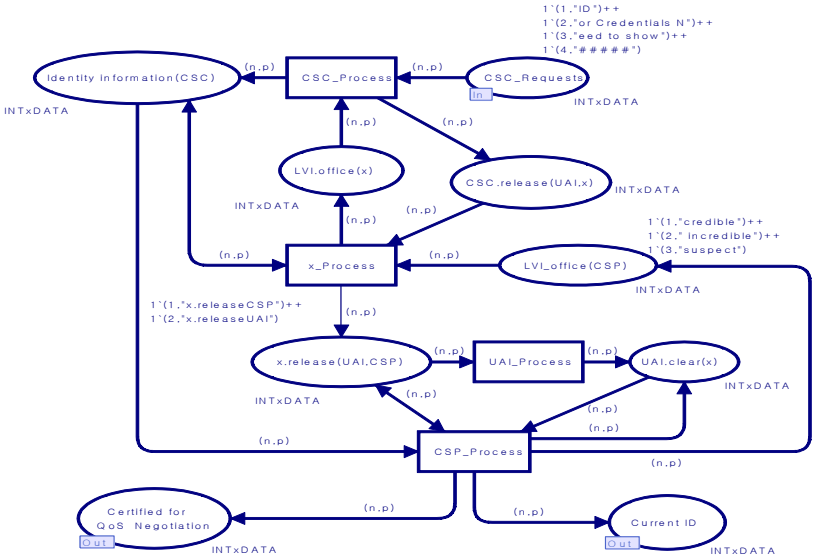

Figure 6. The Sub-diagram of the HCPN Model on Authenticate Identity

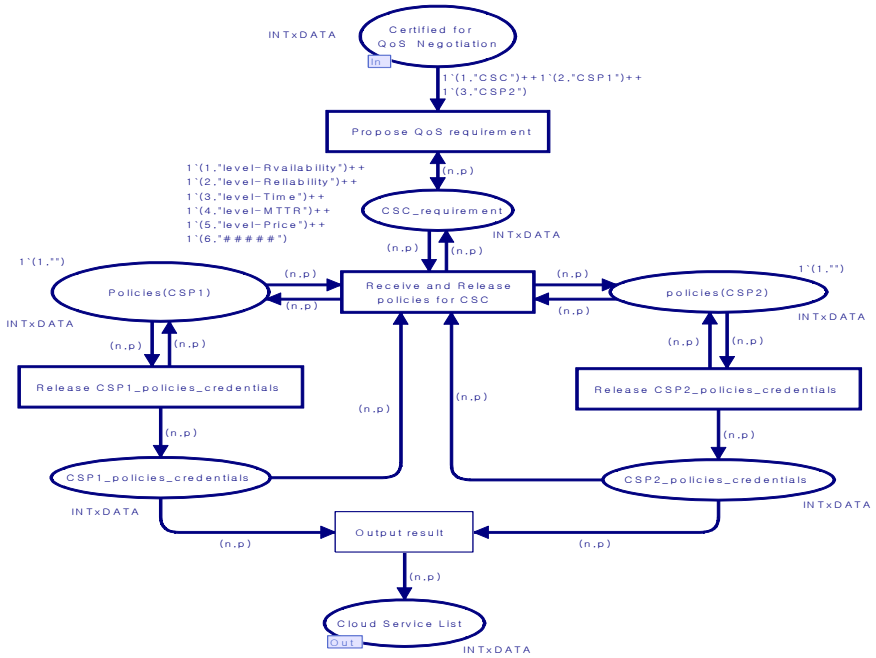

Figure 7. The Sub-diagram of the HCPN Model on QoS Negotiation

In SLA trust negotiation, the CSC puts more important service properties in front consultation according to their own preferences [13, 14], and the price is get in the final negotiations after the service level is determined, so the priority sequence of negotiation in this paper is $Av, R, Re, MTTR, T, P$. In order to shorten the time of the SLA negotiation, CSC negotiates with CSP1 and CSP2 at the same time, and in every round of negotiations,

the CSC releases the request of the service level of the same properties to different cloud service providers at the same time. After the CSP side receives the request, it discloses to CSC the corresponding negotiation strategies of service properties. After the CSC receives the negotiation strategies of the other side, it assesses the reliability or the feasibility of these strategies, and decides whether to disclose trust certificates to the other side. If the trust certificate is disclosed, the next negotiation of service properties is performed, until the other party is notified the end of negotiation after the success of the negotiation price. If the trust certificate is not disclosed, then on the basis of a new proposal put forward by each supplier, the CSC new proposals are generated for the next round of negotiations.

Due to the credibility of R computed by the authentication authorization stage, the second stage only needs to server the kth cloud service for 5 attributes for service level i $QS_k = \{Av_i, Re_i, MTTR_i, T_i, P_i\}$ to negotiate. Figure 8 shows negotiated sequences for QoS negotiation.

Consultation target of this phase is mainly availability (Av_i), reliability (Re_i), the average failure time to repair ($MTTR_i$), and service execution time (T_i). Also, the application function surrounds these negotiation goals.

(1) Suppose that in the example above, SLA sampling time T for SaaS cloud services is 24 hours, and the availability for average time under the time span K refers to the size of the actual ability to provide services in a period of the probability balance for a resource as a service provider. The average length of service availability expression AV_s is as follows.

$$AV_s = \frac{\sum_{i=1}^{K/24} Av(i)}{K/24} = \frac{\sum_{i=1}^{K/24} Av(1) + Av(2) + \ldots + Av(K/24)}{K/24} = \frac{\sum_{i=1}^{K/24} \frac{T_S(i) - T_M(i) - T_B(i)}{T_S(i)}}{K/24} \quad (3)$$

T_S is the service execution time of SLA agreement for a service resource, T_M is the scheduled maintenance time as agreed upon by both parties, including routine outage, release, changes, and T_B is a impact time for breakdown.

(2) The reliability of SaaS cloud services mainly refers to the probability of cloud computing system not being failure to provide regulation of cloud services within the time stated in the SLA and given the operating conditions, and reliability can be measured by MTTR. If the interval of current time to next failure time is ξ with a cumulative probability density function $F(t) = P(\xi \le t)$, then the reliability function is

$$R(t) = 1 - F(t) = P(\xi > t) \quad (4)$$

In the running cycle, the expected time between current time and the next failure time of cloud services $E(t)$ reflects the size of $MTTR$ [15].

$$E(t) = \int_0^\infty tf(t)dt = \int_0^\infty R(t)dt = \int_0^\infty [1 - F(t)]dt \quad (5)$$

From Formula (4) and (5), it can be seen that, the more the reliability degree is, the higher the reliability; MTTR being smaller means that the higher the reliability of the system is, the higher the reliability of cloud services is.

(3) The used price of the CSC P_s: the user's billing way is to rent servers with fee P_{chour} per hour per server, and the user rental price calculated by Formula (6).

$$P_s = \sum_{h=1}^{n} (P_{chour}^{ij} \times T_h) \times \delta \quad (6)$$

In which δ means the adjustment parameters of prices in the process of negotiations; P_{chour}^{ij} is the price per hour for a business of grade i and type of business as j; T_h is a billing cycle, and n means the number of real-time billing cycle.

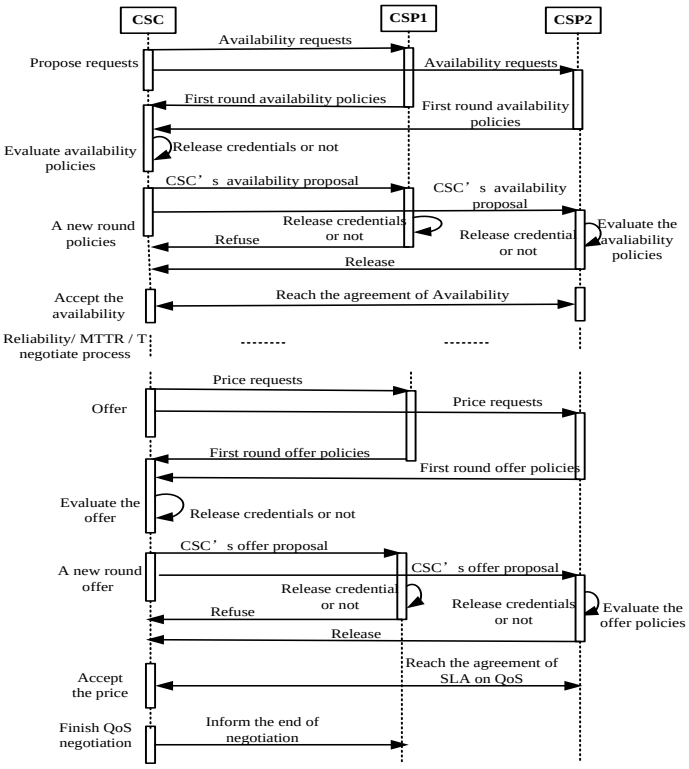

Figure 8. The Sequence Diagram of QoS Negotiation

5. The Multi-objective Optimization Algorithm for SLA Trust Negotiation

5.1. The Multi-objective Optimization Function

The above example is a multiparty trust negotiation on SLA, and it can be divided into two concurrent bilateral trust negotiations such as $\{(CSC,CSP1),(CSC,CSP2)\}$. Eventually, the CSC chooses a globally optimal CSP whose QoS properties meet the requirements, and signs an SLA agreement with the CSP. The QoS negotiation is a negotiation with multiple properties, so it can be converted into an SLA trust negotiation with multi-objective optimization [16-17]. From the CSC perspective, the multi-objective SLA negotiation aims to produce a set of indexes which can meet the CSC's requirements. This paper involves the objective functions such as cedibility(R), price(P), availablity(Av_i) and reliability(Re_i). In addition, time T_s and MTTR are also reflected in the objective functions.According to the above established CPN model and Formula 1-6, the multi-objective optimization problem can be described as the following formula.

$$\begin{cases} objective1: F_1(S) = \sum_{i}^{|m|} \alpha_1 F(C_i) + \alpha_2 g(C_{dis}) / |C_{need}| + \alpha_3 F_{i,act} \\ objective2: F_2(S) = Av_s = \dfrac{\sum_{i=1}^{K/24} Av(i)}{K/24} = \dfrac{\sum_{i=1}^{K/24} \dfrac{T_S(i) - T_M(i) - T_B(i)}{T_S(i)}}{K/24} \\ objective3: F_3(S) = Re_s == \dfrac{-R'(t)}{R(t)} = \dfrac{-(1-F(t))'}{1-F(t)} \\ objective4: F_4(S) = P_s = P_s = \sum_{h=1}^{n} (P_{chour}^{ij} \times T_h) \times \delta \end{cases} \quad (7)$$

Each objective function can be given the weights for the multi-objective optimization problem, and then the comprehensive objective function is defined as:

$$F = \sum_{k=1}^{4} F_k(S)\omega_k \tag{8}$$

in which $\sum_{i=k}^{4} \omega_k = 1, \omega_i \in [0,1]$. ω_1 , ω_2 , ω_3 , ω_4 represents the weight value of the four objective functions. As the unit of various objective functions is inconsistent, they need to be normalized, and can expressed as $F^*_1(S)$, $F^*_2(S)$, $F^*_3(S)$, $F^*_4(S)$. Then, the processed comprehensive objective function is

$$F^* = \omega_1 \frac{F_1(S)}{F^*_1(S)} + \omega_2 \frac{F_2(S)}{F^*_2(S)} + \omega_3 \frac{F_3(S)}{F^*_3(S)} + \omega_4 \frac{F_4(S)}{F^*_4(S)} \tag{9}$$

5.2. The Multi-objective Optimization Algorithm

It is essential for the CSC to choose a globally optimal CSP whose QoS properties meet the requirements. The genetic algorithm (GA) is a search heuristic that mimics the process of natural selection [18]. We can utilize the genetic algorithm to solve the multi-objective optimization problem. In this paper, an improved genetic algorithm is combined with the HCPN model to optimize the multi-objective problem for QoS-based SLA negotiation. Figure 9 is a flow chart of the multi-objective optimization algorithm for SLA trust negotiation, and the specific explanation is as follows.

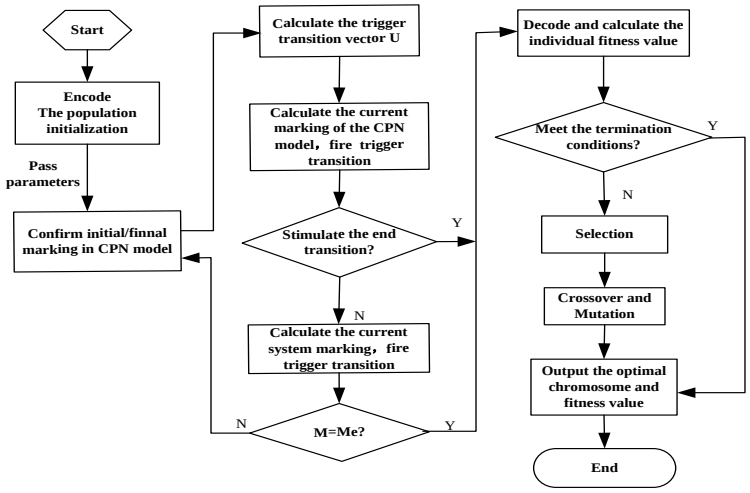

Figure 9. The Flow Chart of the Multi-objective Optimization Algorithm

(1) Encoding for the HCPN model. The multi-objective optimization problem for QoS-based SLA negotiation can be expressed as a set of parameters, which are linked together to form a chromosome. According to the nature of the problem, when using the method of piecewise coding to construct chromosome, each part corresponds to the type of a negotiation strategy. Gene encoding scheme based on HCPN models can be divided into two parts. The first part is a strategy genetic form for the SLA negotiation model, which is the combination of rules and trusts for the multilateral negotiation parties. The second part is the genetic form of parameter weighting for consultation of service properties, which corresponds to a SLA agreement in the SLA negotiation model, composed of two parts as an overall solution for SLA multiparty negotiations.

In these two parts, respectively convert each part of the chromosome coding into the change excitation sequence of Petri net, and determine the color of the library. In colored Petri net, vectors to trigger change can be represented as $\mathbf{U} = [\mathbf{U}_1, \mathbf{U}_2, \ldots \mathbf{U}_n]^T$, if the change T_j is triggered, then $U_j = 1$. There is an algebraic relationship after change

as $M(k) \xrightarrow{\sigma} M(q) = M(k) + A \bullet U$, in which A is the incidence matrix, $A_{i,j} = W(T_j, P_i) - W(P_i, T_j)$, and $M(q)$ is sequence identification after the change sequence σ , or namely the identification of current state. The current identification $M(q)$ and the condition matrix Q can trigger the condition of change vector U(k+1): $M(k) + QU(k+1) \geq 0$, in which assume that the colored Petri net termination is identified as Me , shown as the model in Figure 7. When the cloud service list in the library has token, and other libraries do not have token, the SLA negotiation is completed, Then HCPN reaches the end state, and the terminating identification at this time is $Me = \{0,0,0,0,0,0,1\}$. When $M = Me$, the HCPN stops running.

(2) Fitness value [19 to 20]. The individual fitness value reflects the merits of the individual ability to adapt to the environment, and it is the superior standard to assess individuals in the genetic algorithm. The goal of this paper is to study the problem to optimize service properties of multiple consultations, in order to make cloud users in multiple cloud service providers to choose the QoS with the best global optimization, and sign the SLA document. So the value of the fitness function is set as the target weighted to get an adaptive value $fitness = \omega_1 f(1) + \omega_2 f(2) + \omega_3 f(3) + \omega_4 f(4)$, and it is also the objective function through consultation $F^*(S)$.

(3) Improved genetic operator. The choice is to directly copy the individual of the highest adapting value in a population to the next generation or exclude the bad individual. This paper adopts the roulette-wheel selection mechanism, then the probability of each chromosome been selected is proportional to its fitness. The one with higher fitness has higher probability being selected. The purpose of the cross is to obtain excellent individuals, and interchange some parts of chromosomes (string), resulting in a new individual. The range of crossover probability values is usually (0.5, 1). The variation is used to produce new individual, in order that the genetic algorithm has the local search ability and maintains the diversity of population, with 0 to 1 and 1 to 0 after mutation, The range for mutation probability values is (0, 0.05), and the character position with mutation is randomly generated.

In order to improve the disadvantage that the genetic algorithm is easy to premature convergence and into the local convergence, adopt the evolutionary stable strategy [21], in which every group evolves only replacing two individuals with the worst fitness and individual evolutionary direction is to refer the metropolis standards of simulated annealing algorithms, and accept the deteriorating solution at a certain probability, in order to make the algorithm jump from local optimum and find the global optimal solution.

$$A = \begin{cases} 1, & fitness(i) \geq fitness(j) \\ \exp\left(\dfrac{fitness(i) \geq fitness(j)}{T}\right), & fitness(i) < fitness(j) \end{cases} \quad (10)$$

When randomly selecting from new-generation children individual i of parent individual j , if $fitness(i) \geq fitness(j)$, then the individual i with probability 1 is included and j is eliminated, in which T is the control parameter for annealing process.

5.3. Simulation Results and Analysis

The simulation parameters are as follows: the multi-objective optimization algorithm is on the basis of metropolis criterion; the fitness function is $fitness = 1/F^*$, and $\omega_1, \omega_2, \omega_3, \omega_4$ are set to 0.35, 0.28,0.25,0.12; the initial population are set as $popsize = 10$, the maximal number of generation is 500, the crossover probability is 0.85, and the probability of mutation is 0.01, and T=0.7.

Figure 10 is the contrast curve diagram of the largest fitness and the average fitness. The average fitness can reflect dominant population and inferior population. Parents can

choose certain proportion in the two groups from the average fitness curve, so individuals with disadvantages also have the opportunity to participate in evolution. In general, the average fitness is gradually increasing.

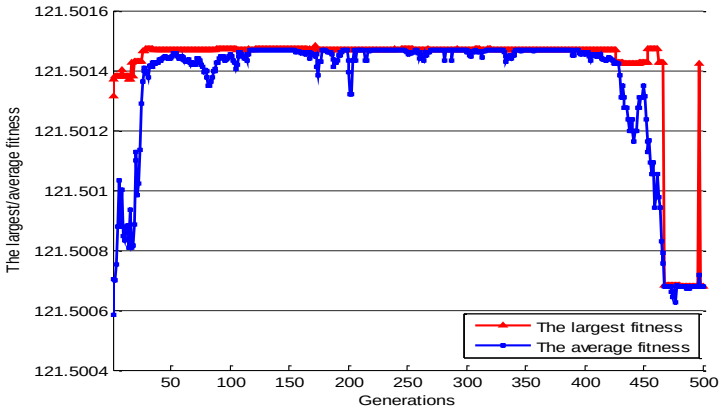

Figure 10. The Contrast Curve Diagram of the Largest Fitness and the Average Fitness

Figure 11 is the contrast curve diagram of Simple Genetic Algorithm (SGA) and Metropolis based Genetic Algorithm (MBGA) strategies on QoS negotiation. We get simulation results 1000 times from four objective functions of tripartite SLA negotiation with these two methods respectively, and calculate fitness D_{CSC} , D_{CSP1} , D_{CSP2} , D . D is the overall fitness, and carries on the quantitative between [0, 1]. When D >0.5 and D_{CSP1} , D_{CSP2} , D_{CSC} are all greater than 0.7, the negotiation is successful.

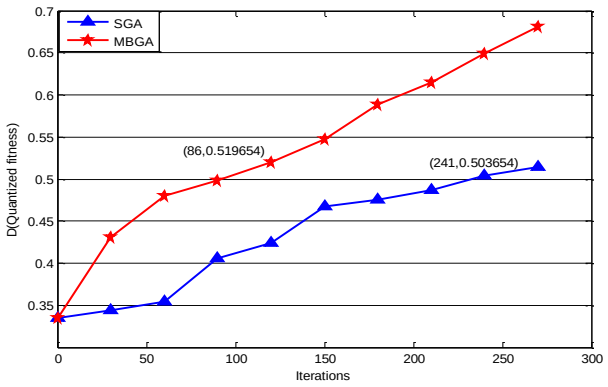

Figure 11. The Contrast Curve Diagram of SGA and MBGA Strategies on QoS Negotiation

6. Conclusions

The characters of the cloud computing environment such as openness, dynamics and complexity bring trust problems. In addition, The CSC faces with multiple selections of CSPs before its cloud service. The multiparty trust negotiation on SLA can solve the problems above. This paper presents an SLA-oriented multiparty trust negotiation model based on HCPN in cloud environment, which can simplify the negotiation process effectively. Meanwhile, the output results of the model can convert a multi-objective SLA negotiation problem. The CSC can choose an optimal CSP and signs the SLA agreement with the CSP. The simulation results show that the multi-objective optimization algorithm on SLA negotiation is feasible. The algorithm adopts the metropolis criterion in respect of

the population evolution, so it can effectively speed up the negotiation efficiency and avoid the local optimum problem. In the future, we would further study the many-to-many trust negotiation problem on SLA in cloud environment.

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).

References

[1] J. Ortiz, V. T. de Almeida and M. Balazinska, "A vision for personalized service level agreements in the cloud", Proceedings of the Second Workshop on Data Analytics in the Cloud, ACM, **(2013)**, pp. 21-25.
[2] S. H. Zegordi and H. Davarzani, "Developing a supply chain disruption analysis model: Application of colored Petri-nets", Expert Systems with applications, vol. 39, no. 2, **(2012)**, pp. 2102-2111.
[3] O. Farras, J. Domingo-Ferrer and A. Blanco-Justicia, "Privacy-preserving trust management mechanisms from private matching schemes", Data Privacy Management and Autonomous Spontaneous Security, Springer Berlin Heidelberg, **(2014)**, pp. 390-398.
[4] L. Wu, S. K. Garg and R. Buyya, "SLA-based admission control for a Software-as-a-Service provider in Cloud computing environments", Journal of Computer and System Sciences, vol. 78, no. 5, **(2012)**, pp. 1280-1299.
[5] M. B. Chhetri, Q. B. Vo and R. Kowalczyk, "Policy-based automation of SLA establishment for cloud computing services", 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid). IEEE, **(2012)**, pp. 164-171.
[6] C. Stahl, M. Westergaard and W. M. P. van der Aalst, "Strategies for modeling complex processes using colored petri nets", Transactions on Petri Nets and Other Models of Concurrency VII, Springer Berlin Heidelberg, **(2013)**, pp. 6-55.
[7] Y. Cardinale, J. El Haddad, M. Manouvrier, M. Rukoz, "CPN-TWS: a coloured petri-net approach for transactional-QoS driven Web Service composition", International Journal of Web and Grid Services, vol. 7, no. 1, **(2011)**, pp. 91-115.
[8] Y. B. Mustapha and H. Debar, "Service Dependencies-Aware Policy Enforcement Framework Based on Hierarchical Colored Petri Net", Security in Computing and Communications, Springer Berlin Heidelberg, **(2013)**, pp. 313-321.
[9] S. Jing-zhou, M. Tie-jun, S. Han-xu, J. Qing-Xuan and G. Xin, "Modeling of augmented reality assembly system based on hierarchy colored Petri net", Computer Integrated Manufacturing Systems, vol. 18, no. 10, **(2012)**, pp. 2166-2174.
[10] P.-J. Chuang and M.-Y. Ni, "On access control policy assignments and negotiation strategies in automated trust negotiation", International Journal of Security and Networks, vol. 9, no. 2, **(2014)**, pp. 104-113.
[11] Y. Li and Y. Liu, "Research on Modeling of Multiparty Trust Negotiation Based on Coloured Petri-net in P2P Network", Second International Conference on Networks Security Wireless Communications and Trusted Computing (NSWCTC), IEEE, **(2010)**, pp. 437-441.
[12] C. C. Zhang and M. Winslett, "Distributed authorization by multiparty trust negotiation", Computer Security-ESORICS 2008, Springer Berlin Heidelberg, **(2008)**, pp. 282-299.
[13] J. Conejero, B. Caminero, C. Carrión and L. Tomás, "From volunteer to trustable computing: Providing QoS-aware scheduling mechanisms for multi-grid computing environments", Future Generation Computer Systems, vol. 34, no. 5, **(2014)**, pp. 76-93.
[14] L. Blasi, J. Jensen and W. Ziegler, "Expressing Quality of Service and Protection Using Federation-Level Service Level Agreement", Euro-Par2013: Parallel Processing Workshops, Springer Berlin Heidelberg, **(2014)**, pp. 146-156.
[15] M. A. Chalouf, N. Mbarek and F. Krief, "Quality of Service and security negotiation for autonomous management of Next Generation Networks", Network Protocols and Algorithms, vol. 3, no. 2, **(2011)**, pp. 54-86.
[16] K. Deb, "Multi-objective optimization", Springer US in Search methodologies, **(2014)**, pp. 403-449.
[17] F. Wang, X. Lai and N. Shi, "A multi-objective optimization for green supply chain network design", Decision Support Systems, vol. 51, no. 2, **(2011)**, pp. 262-269.
[18] R. V. Rao and V. Patel, "Multi-objective optimization of heat exchangers using a modified teaching-learning-based optimization algorithm", Applied Mathematical Modelling, vol. 37, no. 3, **(2013)**, pp. 1147-1162.
[19] M. O. Shabani and A. Mazahery, "Application of GA to optimize the process conditions of Al Matrix nano-composites", Composites Part B: Engineering, vol. 45, no. 1, **(2013)**, pp. 185-191.

[20] A. S. Tasan and M. Gen, "A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries", Computers & Industrial Engineering, vol. 62, no. 3, **(2012)**, pp. 755-761.
[21] Q. Zhao and H. Chen, "A Supplier's Optimal Pricing Strategy and Evolutionarily Stable Strategies of Retailers with Different Behavior Rules", International Journal of Digital Content Technology and its Applications, vol. 6, no. 12, **(2012)**, pp. 440-448.

Authors

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

Qiuxia Chen, She is from Guangxi Province of China, and a master candidate at School of Computer Science in Hubei University of Technology, interested in Cloud Computing.

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 an associate 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 Peer-to-Peer Computing, Cloud Computing and SDN.

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.