Configuration and Optimization of Virtual Business in Cloud Computing Environment

Hongjun Sun
China Aerospace Engineering Consultation Center
Beijing, PR China
sunhj05@sina.com

Shuangxi Huang*, Yushun Fan and Wei Su
Department of Automation, Tsinghua University
Beijing China
{huangsx, fanyus, weisu2012}@tsinghua.edu.cn

***Abstract*—Cloud computing enables the realization of virtual business through outsourcing business functions to online services. Due to the large number of available business services and candidate cloud service providers, the configuration and optimization of the virtual business becomes a key issue for a service-oriented business ecosystem. In this paper, the virtual business is regarded as a series of service processes. Constrained service workflow net (CSWF-net) is developed to describe the component and structure of such service processes. Considering the dynamic and stochastic nature of services in a cloud computing environment, a stochastic decision making approach for CSWF-net is proposed in which the state transition of services is characterized as a Continuous Time Markov Chain (CTMC). Based on the four control structures of service processes, the dynamic Quality-of-Service (QoS) aware selection and composition problem is formulated as a Stochastic Linear Programming (SLP) model. Mixed policies generated by the branch and bound algorithm and modified heuristic approach are used to solve the dynamic QoS-aware selection and composition problem at runtime. In the meantime, the most appropriate business services and cloud service providers can be selected to support the execution of the virtual business effectively.**

Keywords-Service processes; service workflow net; continuous time markov chain; stochastic linear programming

I. INTRODUCTION

With the advancement in services computing and cloud computing, more and more enterprises offer their resources and business processes as services through Internet. This whole collection of business services is referred as a Service-oriented Business Ecosystem(SOBE)[1]. SOBE offers a realization of establishing and running a virtual business, in which most of its business functions are outsources to online services [2]. Through the selection and composition of services, the virtual business can be modeled and realized as service processes. In execution, service processes needs to be separated into small partitions and allotted to different cloud service providers to execute [3].

Presently, researchers have gotten many achievements in service process partitioning for distributed execution. Using Business Process Execution Language (BPEL), IBM India research center carries out research on distributed model fragmentation, the validation of the fragmentation, and data flow constraint based orchestration [4]. Liu proposed a role-based decentralized approach for service composition[5]. In [6], based on Petri nets, static and dynamic workflow model partitioning approaches are introduced, the structure correctness is validated and the partition policy and efficiency are analyzed preliminarily. However, as far as we have known, the researchers don't pay enough attention to how to align cloud service providers for the execution of fragments and ignore some important factors which have important influence on the execution performance of service processes at runtime.

Because there are a large number of similar or equivalent business which can be provided by different cloud service providers. It is thus a key issue to select appreciate business services and cloud service providers to meet business requirements in economic and effective way.

Most of the existing researches in service selection and composition are based on quality-of-service constraints. Menascé examined how quality of service (QoS) comes into play for service providers, consumers, and parallel transactions[7]. Liu presented a global optimal algorithm of dynamic Web services selection to resolve dynamic Web services selection with QoS global optimal in Web services composition[8]. Li et al. proposed a high performance approach for multi-QoS constrained Web Services selection [9]. Huang designed an optimal QoS-based service selection scheme and proposed multiple criteria decision making and integer programming approaches to select the optimal services[10].

However, It is not sufficient for service process execution in cloud computing environment to consider the QoS characteristics of the services only. The cloud computing environment and the performance for each cloud service provider also need to be considered.

Furthermore, the stochastic nature of business services and the dynamic nature of cloud computing environment are unavoidable [11]. The dynamic and stochastic nature of services and cloud computing environment play an important role in execution performance of service processes.

In this paper, a multi-level service workflow model, named Constrained Service Workflow net(CSWF-net), is proposed to describe the structure of service processes. CSWF-net extended existing service workflow net with multiple constraints influencing service process execution performance. Moreover, role is considered to build the relationship between any pair of transitions, business services and cloud service providers. Considering the environment dynamism, the state transition of service is described as Continuous Time Markov Chain (CTMC). Based on the four basic structures of service processes, the dynamic selection and composition problem is described as Stochastic Linear Programming (SLP) model. The mixed

978-0-7695-4864-7/12 $26.00 © 2012 IEEE
DOI 10.1109/CGC.2012.73

policies based on branch and bound algorithm and modified heuristic algorithm are considered to solve the problem at runtime. The advantages of the proposed methods include the enhanced performance by considering the factors influencing service process execution, the increased flexibility by designating cloud service providers at runtime, the avoidance of concurrent execution bottleneck of activities by concurrent execution constraint, etc.

II. CONSTRAINED SERVICE WORKFLOW NET

In cloud computing environment, the execution performance and efficiency of service processes are under the great influence of some factors. These constraints can be divided into two types: hard constraint and soft constraint. The hard constraint is the forcible constraint that model fragments must follow, including organization barrier, resource capability, data security, etc. The soft constraint is the non-forcible constraint, such as user preference, execution concurrency, data transfer.

Considering these factors influencing service process execution synthetically, Constrained Service Workflow net (CSWF-net) is defined by extending traditional Workflow Net(WF-net).

WF-net is a special class of Petri net, which prevails in workflow modeling field because of its graphic nature and theoretical foundation [12]. It is suitable not only for representation and validation, but also for the verification of business processes. The Service Workflow net(SWF-net) is a kind of WF-net that represented by a colored Petri net with structure characteristics of a WF-net[3].

Definition 1: Constrained Service Workflow net (CSWF-net) is a quadruple (W, Rl, C, R), where:

(i) $W = \{W_1, W_2, \cdots, W_u\}$ is the set of SWF-nets;

(ii) $Rl = \{Rl_1, Rl_2, \cdots, Rl_m\}$ is roles set, where Rl_i for $i = 1, 2, \cdots, m$ is a role defined in business context;

(iii) $C = \{Rc, O, S\}$ is constraints set, where:

(a) $Rc = \{Rc_1, Rc_2, \cdots, Rc_n\}$ is business service pools set, where Rc_i for $i = 1, 2, \cdots, n$ is a business service pool;

(b) $O = \{O_1, O_2, \cdots, O_l\}$ is cloud service providers set, where O_i for $i = 1, 2, \cdots, l$ represents a cloud service provider;

(c) $S = \{Su, Sd, Sc, \cdots\}$ is soft constraints set, where Su represents user preference corresponding to each transition, Sd represents data transfer, and Sc represents concurrent execution. Other constraints can be extended in this model.

(iv) $R = \{Rr, Rs, Ro, Rp\}$ is relations set, where:

(a) $Rr : T_{W_i} \to Rl$ represents the mapping relations between W and Rl, where T_{W_i} is a transitions set of SWF-net W_i. For $\forall T_i \in T$, the roles set $subRl \subseteq Rl$ is needed to execute T_i, where role $Rl_i \in subRl$ takes charge of the execution of T_i as the charge role;

(b) $Rs \subseteq Rl \times Rc$ represents the binary relations between Rl and Rc. For $\forall T_i \in T$, the charge role Rl_i is needed to deploy business services from appointed business service pools for the firing of transition T_i;

(c) $Ro \subseteq Rl \times O$ represents the many to many relations between Rl and O. For $\forall Rl_i \in Rl$, Rl_i corresponds to more collaboration services, and $\forall Rc_i \in Rc$, Rc_i may correlate more roles;

(d) $Rs : T_{W_i} \to S$ represents the mapping relations between the transitions T_{W_i} and the soft constraint S.

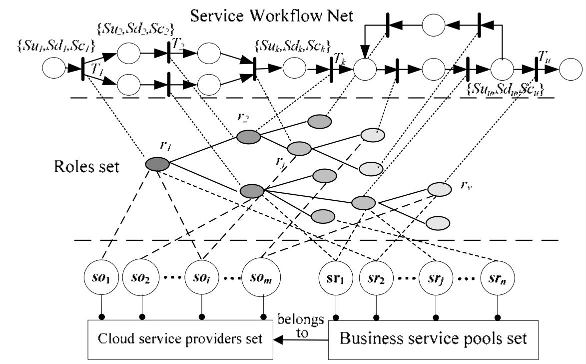

Figure 1. CSWF-net graphic representation.

Figure 1 shows the graphic representation of the constrained service workflow net. In the first level, service workflow net is specified, which has a soft constraints set $\{Su_k, Sd_k, Sc_k\}$ for transition T_k. Here, Su_k represents user preference constraint, Sd_k represents data transfer constraint, and Sc_k denotes concurrent execution constraint. The second level is the role perspective, in which each transition T_k is appointed a specific role r_j for its firing. In the third level, all the business services pools capable of performing the roles that are described in the second level. Furthermore, cloud service providers that business services pools belong to are illustrated, which are the cloud service providers of service execution. Here, so_i for $i = 1, 2, \cdots, m$ represents cloud service provider, and sr_j for $j = 1, 2, \cdots, n$ represents business services pool. The dotted lines describe the mapping relations among the three perspectives.

CSWF-net is the extension of WF-net, and has the basic properties that WF-net has. The CSWF-net is sound if and only if (P, T, F) is a sound WF-net. Here, we only consider sound CSWF-net. According to service process fragmentation algorithm proposed in [3], CSWF-net can be separated into parts dynamically at runtime.

Considering the basic control structures of service processes [13], the execution structure of the fragments can be divided into four kinds: sequence execution fragment, concurrent execution fragment, alternative execution fragment and iterative execution fragment.

The arriving transaction instance in the first level of CSWF-net can be constrained by soft constraints directly and can be mapped into the second level. Based on the second level, the mapping between transaction instance and business services is built. Based on the four execution fragments, the following sections show how to configure and optimize service processes based on the structures and mapping relations in CSWF-net.

III. SERVEICE PROCESSES STOCHASTIC DECISION MODEL BASED ON CSWF-NET

In this section, we first acquire the cloud service provider candidates for each fragment based on the relation between any pair of constraint, role, and transition described in CSWF-net. Then, according to the control structure for each fragment, the QoS-aware service processes stochastic decision problem is modeled as stochastic linear programming problem for multi objectives.

A. Cloud Service Providers Candidates Acquirement

Given a CSWF-net W , $T=\{T_1,T_2,\cdots,T_m\}$ represents its transitions set, $Rl=\{Rl_1,Rl_2,\cdots,Rl_n\}$ represents the roles set, $O=\{O,O_2,\cdots,O_v\}$ is the cloud service providers set, and $Rp=\{Rp_1,Rp_2,\cdots,Rp_\mu\}$ represents the business service pools set. According to the relation defined in W , we can get the following definitions.

In CSWF-net, roles take charge of the execution of services and each transition corresponds to a leader role. The mapping relation $F:T\rightarrow R$ in CSWF-net specifies distinctly the relations between roles and transitions.

Definition 2: According to the mapping relation $F:T\rightarrow R$ in CSWF-net W , the transition vs. role matrix $M=\{m_{ij}\}_{m\times n}$ can be defined as an $m\times n$ matrix, where $m_{ij}=0$ or 1.

Here, $m_{ij}=1$ represents that the firing of T_i need the support of Rl_j, and $m_{ij}=0$ is the opposite. Since the firing of each transition generates a request to a specific role, there is only one item with the value 1 in each column of the matrix M_k .

Each role can invoke business services from multi cloud service providers, and each cloud service provider includes multiple roles. According to the many to many relations $Ro\subseteq Rl\times O$ in CSWF-net W , the role vs. cloud service provider matrix can be constructed.

Definition 3: The role vs. cloud service provider matrix $N=\{n_{ij}\}_{n\times v}$ can be defines as an $n\times v$ matrix where $n_{ij}=0$ or 1 based on the relation $Ro\subseteq Rl\times O$ in W .

Here, $n_{ij}=1$ represents that O_j contains Rl_i , and $n_{ij}=0$ is the opposite. When $n_{ij}=1$, there is one or more business service pools candidate to support the execution of role Rl_i in cloud service provider O_j .

Definition 4: According to the binary relation $Rs\subseteq Rl\times Rc$ defined in CSWF-net W , the role vs. available business service pools matrix $A=\{a_{ij}\}_{n\times v}$ can be defined as an $n\times v$ matrix where $a_{ij}=0$ or 1.

Definition 5: For $\forall Rl_i,Rp_j$, assuming that Rp_j belongs to O_k , if $a_{ij}=1$ and $n_{ik}=0$, then $a_{ij}=0$, $a_{ij}'=a_{ij}$. The role vs. business service pools matrix $A'=\{a_{ij}'\}_{n\times v}$ is an $n\times v$ matrix where $a_{ij}'=0$ or 1.

When $a_{ij}'=1$, business service pool Rp_j supports role Rl_i, and when $a_{ij}'=0$, it is the opposite.

According to definition 2 and definition 3, the transition vs. cloud service provider matrix $B=\{b_{ij}\}_{m\times v}=M^T\times N$ can be represented as an $m\times v$ matrix where $b_{ij}=0$ or 1. Here, $b_{ij}=1$ represents that O_j is the candidate service of T_i , and $b_{ij}=0$ is the opposite.

Furthermore, based on definition 2 and definition 5, the transition vs. business service pools matrix $A''=M^T\times A'=\{a_{ij}''\}_{m\times v}$ is defined as an $m\times v$ matrix where $a_{ij}''=0$ or 1. Accordingly, for each transition in CSWF-net W , the cloud service providers and business services candidates can be constructed through the above process.

In the following section, we will introduce a stochastic decision approach to build up the service processes by selecting appropriate cloud service providers for each fragment at runtime.

B. Problem Specification For Stochastic Decision

Given a sequential execution fragment F_s that corresponds to a single activity waiting for execution, $X=\{X_1,X_2,\cdots,X_{l_S}\}$ represents its business service candidates set with l_s candidate business services, $\hat{\xi}_i=\{\hat{\xi}_{i1},\hat{\xi}_{i2},\hat{\xi}_{i3},\hat{\xi}_{i4}\}$ is the QoS metric value of X_i for $i=1,2,\cdots,l_s$ and it is a four dimensional random vector, $Y=\{Y_1,Y_2,\cdots,Y_{l_S}\}$ represents the cloud service providers set that X belongs to, $X_j\in Y_j$, $\hat{\gamma}_i=\{\hat{\gamma}_{i1},\hat{\gamma}_{i2},\hat{\gamma}_{i3},\hat{\gamma}_{i4}\}$ is the QoS metric value for Y_i for $i=1,2,\cdots,l_s$ and it is a four dimensional random vector too. Furthermore, $\hat{\xi}_{i1}$ represents the execution time random variable for the candidate service X_j , $\hat{\xi}_{i2}$ represents its cost random variable for X_j , $\hat{\xi}_{i3}$ represents the availability random variable caused by the state transition of business service X_j, and $\hat{\xi}_{i4}$ represents the reliability random variable. And $\hat{\gamma}_{ik}$ for $k=1,2,3,4$ represent the QoS metric values for Y_i, which corresponds to time, cost, availability, reliability random variables respectively.

For the concurrent execution fragment F_c which has l_c

branches to execute concurrently, we assume that there are $l_c^{'}$ branches to select service from the same business services candidates.Let $Z=\{Z_1,Z_2,\cdots,Z_{l_c}\}$ represent the business services candidates set for the $l_c^{'}$ branches, $X=\{X_1,X_2,\cdots,X_{l_c^{'}}\}\ (X\subseteq Z)$ represent the selected business services for the $l_c^{'}$ branches, and $Y=\{Y_1,Y_2,\cdots,Y_{l_c^{'}}\}$ represent the cloud service providers set that X belongs to, $\hat{\xi}=\{\hat{\xi}_1,\hat{\xi}_2,\cdots,\hat{\xi}_{l_c^{'}}\}$ is the QoS metric value for X and it is an $l_c^{'}\times 4$ random matrix, $\hat{\gamma}=\{\hat{\gamma}_1,\hat{\gamma}_2,\cdots,\hat{\gamma}_{l_c^{'}}\}$ is the QoS metric value for Y and it is an $l_c^{'}\times 4$ random matrix.

Furthermore, Q_t represents time threshold, Q_c is cost threshold, Q_a is availability threshold, and Q_r is reliability threshold. The dynamic service selection for selectable and iterative execution fragments can reference the sequential and concurrent execution fragments. Finally, the dynamic configuration and optimization problem of service processes can be formularized into SLP problems as follows.

C. *Service processes stochastic decision under multiple QoS objectives*

If the customers have multiple QoS objectives, the configuration and optimization problem of service processes can be treated as Multi-Attribute Decision Making (MADM) [14], in which alternatives are the candidate business services for each fragment and attributes are the QoS measures of the candidate business services under consideration. Simple Additive Weighting (SAW) [15] is applied to obtain the aggregated value of QoS, which is widely used to obtain a score from a list of dimensions. In cloud computing environment, the uncertainty is due to the fact that performance evaluations of alternatives on each of the attributes lead to random variables with probability distribution. Therefore, the stochastic characteristics also need to be considered in the configuration and optimization of service processes.

Different QoS attribute values may be not at the same quantity level. To balance the difference between measurement units of different QoS metrics, the values need to be normalized to be in the range [0, 1]. For transition T, its business service candidates set is denoted as $R=\{R_1,R_2,\cdots,R_i,\cdots,R_k\}$, $\hat{\xi}_{ij}$ for $j=1,2,3,4$ represents the jth QoS metrics for service R_i, and $\hat{\psi}_{ij}$ represents the normalized result for $\hat{\xi}_{ij}$. For negative criteria, values are scaled according to (1a). And for positive criteria, values are scaled according to (1b).

$$\hat{\psi}_{ij}=\begin{cases}\dfrac{\max\limits_{k=1,2,\cdots,n}(\hat{\xi}_{kj})-\hat{\xi}_{ij}}{\mathrm{E}} & (\mathrm{E}\neq 0)\\ 1 \quad (\mathrm{E}=0)\end{cases} \tag{1a}$$

$$\hat{\psi}_{ij}=\begin{cases}\dfrac{\hat{\xi}_{ij}-\min\limits_{k=1,2,\cdots,n}(\hat{\xi}_{kj})}{\mathrm{E}} & (\mathrm{E}\neq 0)\\ 1 \quad (\mathrm{E}=0)\end{cases} \tag{1b}$$

Where, $\max\limits_{k=1,2,\cdots,n}(\hat{\xi}_{kj})$ and $\min\limits_{k=1,2,\cdots,n}(\hat{\xi}_{kj})$ respectively represent the maximal and minimal value for the jth quality metric, $\mathrm{E}=\max\limits_{k=1,2,\cdots,n}(\hat{\xi}_{kj})-\min\limits_{k=1,2,\cdots,n}(\hat{\xi}_{kj})$.

Then we apply a weight ω_j for the jth QoS attribute to represent relative importance or value trade-offs of different attributes. Finally, we calculate a value $Value=\sum_{j=1}^{q}\hat{\psi}_j*\omega_j$ for each candidate service. Through comparing $Value$ for all candidates, the optimal cloud service provider with the maximal $Value$ is chosen. According to the four execution structures of the fragments, we formulate the service processes decision problem under multiple QoS objectives as follows.

1) The sequential execution fragment

For each sequence execution fragment, $\hat{\psi}_{ij}$ represents the normalized value for $\hat{\xi}_{ij}$, and ω_j is the weight of criterion j. The service processes decision problem can be determined by solving the following formula:

P1. Maximize

$$F(X_j,\hat{\xi}_j,\hat{T}_{rj},\hat{\gamma}_j,\hat{T}_{oj},T_{os},T_c)=\sum_{k=1}^{4}\omega_k\hat{\psi}_{jk} \tag{2a}$$

s.t.

$$\sum_{k=1}^{4}\omega_k=1 \quad \omega_j\in[0,1] \tag{2b}$$

$$\hat{\xi}_{j1}\leq\min(\hat{T}_{rj},\hat{T}_{oj}-T_c-T_{os},Q_t) \tag{2c}$$

$$\hat{\xi}_{j2}\leq Q_c \tag{2d}$$

$$\hat{\xi}_{j3}\hat{\gamma}_{j3}\geq Q_a \tag{2e}$$

$$\hat{\xi}_{j4}\hat{\gamma}_{j4}\geq Q_r \tag{2f}$$

2) The concurrent execution fragment

In order to avoid resource bottleneck in concurrent execution, different branch must select different service. The concurrent execution constraint also needs to be considered. For fragment F_c, assuming that there are l_m concurrent execution branches that select service from the same business services candidates set in the fragment F_c, the cloud service providers selection can be determined by solving the problem P2.

P2. Maximize

$$F(X,\hat{\xi},\hat{\gamma},\hat{T}_{rj},\hat{T}_{oj},T_{os},T_c)=\sum_{i=1}^{4}\omega_i\hat{\psi}_i \tag{3a}$$

s.t.

$$\sum_{i=1}^{4}\omega_i=1 \quad \omega_i\in[0,1] \tag{3b}$$

$$\max_{j=1,2,\cdots,l_m}(\hat{\xi}_{j1})\leq\min(\min_{j=1,2,\cdots l_m}(\hat{T}_{rj}),\min_{j=1,2,\cdots l_m}(\hat{T}_{oj}-T_c-T_{os}),Q_t) \tag{3c}$$

$$\sum_{j=1}^{l_m}\hat{\xi}_{j2}\leq Q_c \tag{3d}$$

$$\prod_{j=1}^{l_m}(\hat{\xi}_{j3}*\hat{\gamma}_{j3})\leq Q_a \tag{3e}$$

$$\prod_{j=1}^{l_m}(\hat{\xi}_{j4} * \hat{\gamma}_{j4}) \le Q_r \quad (3f)$$

$$X_i \ne X_j \ (X_i \in X, X_j \in X, (X_i \to T_i) \& (X_j \to T_j)) \quad (3g)$$

Here, $\hat{\psi}_i = \max_{j=1,2,\cdots,l_m} \hat{\psi}_{ij}$ for $i=1$, $\hat{\psi}_i = \sum_{j=1}^{l_m} \hat{\psi}_{ij}$ for $i=2$, $\hat{\psi}_i = \prod_{j=1}^{l_m} \hat{\psi}_{ij}$ for $i=3,4$, $\min_{j=1,2,\cdots l_m}(\hat{T}_{oj} - T_c - T_{os})$ represents the minimal holding time that the cloud service providers set Y that business services set X belongs to, $\min_{j=1,2,\cdots,l_m}(\hat{T}_{rj})$ represents the minimal failure time of X.

In P2, the reliability and availability constraints can be linearized by applying the logarithm function.

3) The selectable execution fragment

According to the execution result of the former activities, in the selectable execution fragment, the activity that meets the execution condition is selected to execute. The method and formulas in sequence execution structure can be used in the selected branch.

4) The iterative execution fragment

For the iterative execution fragment, the business service needs to be reselected every iteration because of service dynamics. The iterative execution fragment can be regarded as a series of sequence execution activity. The model every iteration can reference P1.

IV. THE SOLUTION

A. Expected Value Model

In order to solve the above stochastic programming problems, the four service processes decision models are transformed into Expected Value Model (EVM) with maximum expected return subject to expected constraints [16]:

$$\begin{aligned} & max\ E[f(X,\hat{\delta})] \\ & s.t. \\ & \quad E[g_j(X,\hat{\delta})] \ge \beta_j \quad j=1,2,\cdots p \end{aligned} \quad (4)$$

Here, X is a decision vector, $\hat{\delta}$ is a random vector with the probability density function $\phi(\hat{\delta})$, f corresponds to the objective function, g_j is the constraint function for $j=1,2,\cdots p$, and $E[\ ]$ is mathematical expectation value operator. The model is called EVM corresponding to the stochastic programming.

In order to get feasible solution efficiently for the EVM, the mathematical expectation for objective function and constrains need to be transformed into certain types.

Markov chains represent the core of most performance and reliability prediction models. The above CTMC based stochastic model formalism can be used to analyze the availability and reliability for a given business service. Furthermore, we assume that service time and cost obey the exponential distribution with the parameter λ_{Tj} for $\hat{\xi}_{j1}$ and λ_{Cj} for $\hat{\xi}_{j2}$. Therefore, $E[\hat{\xi}_{j1}] = 1/\lambda_{Tj}$ and $E[\hat{\xi}_{j2}] = 1/\lambda_{Cj}$. Then,

$$E[\hat{\xi}_2] = E[\sum_{j=1}^{l_m}\hat{\xi}_{j2}] = \sum_{j=1}^{l_m} E[\hat{\xi}_{j2}] = \sum_{j=1}^{l_m} 1/\lambda_{Cj}.$$

Theorem 1: Suppose $\hat{\delta}_1, \hat{\delta}_2, \cdots \hat{\delta}_n$ are independent random variables, and $\hat{\delta}_i\ (i=1,\cdots,n)$ is exponentially distributed with the parameter λ_i denoted as $\hat{\delta}_i \sim E_\delta(\lambda_i)$, then the mathematical expectations for $\min_{1\le i\le n}\hat{\delta}_i$ and $\max_{1\le i\le n}\hat{\delta}_i$ can be written as:

$$E[\min_{1\le i\le n}\hat{\delta}_i] = (\sum_{i=1}^{n}\lambda_i)^{-1} \quad (5a)$$

$$E[\max_{1\le i\le n}\hat{\delta}_i] = \sum_{i=1}^{n}(i\lambda_i)^{-1}. \quad (5b)$$

Proof:

(1) Since

$$\begin{aligned} P(\min_{1\le i\le n}\hat{\delta}_i > T) &= P(\hat{\delta}_1 > T, \cdots, \hat{\delta}_i > T \cdots, \hat{\delta}_n > T) \\ &= P(\hat{\delta}_1 > T) * \cdots * P(\hat{\delta}_i > T) * \cdots * P(\hat{\delta}_n > T) \\ &= \exp(-\sum_{i=1}^{n}\lambda_i)t \end{aligned}$$

That is, $\min_{1\le i\le n}\hat{\delta}_i$ is exponentially distributed, denoted as

$$\min_{1\le i\le n}\hat{\delta}_i \sim E_\delta[\sum_{i=1}^{n}\lambda_i]$$

Then, $E[\min_{1\le i\le n}\hat{\delta}_i] = (\sum_{i=1}^{n}\lambda_i)^{-1}$.

(2) Suppose $\hat{\delta}_{(1)} \le \hat{\delta}_{(2)} \le \cdots \le \hat{\delta}_{(n)}$ represent the order statistics for $\hat{\delta}_1, \hat{\delta}_2, \cdots \hat{\delta}_n$, then the joint probability density function $\phi(\hat{\delta}_{(1)}, \hat{\delta}_{(2)}, \cdots \hat{\delta}_{(n)})$ for $\hat{\delta}_{(1)}, \hat{\delta}_{(2)}, \cdots \hat{\delta}_{(n)}$ can be described as:

$$\phi(\hat{\delta}_{(1)}, \hat{\delta}_{(2)}, \cdots \hat{\delta}_{(n)}) = n!\prod_{i=1}^{n}\phi(\hat{\delta}_{(i)}) = n!\prod_{i=1}^{n}\lambda_i * \exp(-\sum_{i=1}^{n}(\lambda_i)t)$$

Suppose $Z_1 = \hat{\delta}_{(n)} - \hat{\delta}_{(n-1)}$, $Z_2 = 2\hat{\delta}_{(n-1)} - \hat{\delta}_{(n-2)}$, $\cdots$ $Z_{n-1} = (n-1)\hat{\delta}_{(2)} - \hat{\delta}_{(1)}$, $Z_n = n\hat{\delta}_{(1)}$,

Then $\hat{\delta}_{(1)} + \hat{\delta}_{(2)} + \cdots + \hat{\delta}_{(n)} = Z_1 + Z_2 + \cdots, Z_n$

The Jacobi expression is represented as:

$$J = \frac{D[\hat{\delta}_{(1)}, \hat{\delta}_{(2)}, \cdots, \hat{\delta}_{(n)}]}{D[Z_1, Z_2, \cdots, Z_n]} = \frac{1}{n!}$$

Here, $D[]$ represents variance operator. Accordingly, the joint probability density functions $\phi(Z_1, Z_2, \cdots, Z_n)$ for $(Z_1, Z_2, \cdots, Z_n)$ is represented as:

$$\begin{aligned} \phi(Z_1, Z_2, \cdots, Z_n) &= \phi(\frac{Z_n}{n}, \frac{Z_{n-1}}{n-1} + \frac{Z_n}{n}, Z_1 + \frac{Z_2}{2} + \cdots + \frac{Z_n}{n}) \mid J \mid \\ &= \prod_{i=1}^{n}\lambda_i \exp(-\lambda_i Z_i) \end{aligned}$$

Then,

$$\begin{aligned} E[\hat{\delta}_{(n)}] &= E[\max_{1\le i\le n}\hat{\delta}_i] \\ &= E[Z_1] + \frac{1}{2}E[Z_2] + \cdots + \frac{1}{n}E[Z_n] \\ &= \sum_{i=1}^{n}(i\lambda_i)^{-1} \end{aligned}$$

Therefore, Theorem 1 holds. □

Due to the Markov property, the holding time for each state is exponentially distributed. According to Theorem 1, we can get the following formulations:

$$E[\hat{\xi}_1] = E[\max_{j=1,2,\cdots,l_m}(\hat{\xi}_{j1})] = \sum_{j=1}^{l_m}(j\lambda_j)^{-1}$$

$$E[\min_{j-1,2,\cdots l_C}(\hat{T}_{oj} - T_c - T_{os})] = E[\min_{j-1,2,\cdots l_c}(\hat{T}_{oj})] - T_c - T_{os} = (\sum_{j=1}^{l_c}\mu_{Oj})^{-1} - T_c - T_{os}$$

$$E[\hat{T}_r] = E[\min_{j-1,2,\cdots l_C}(\hat{T}_{rj})] = (\sum_{j=1}^{l_c}\mu_{Rj})^{-1}$$

$$E[\hat{T}_{rj}] = (\mu_{Rj})^{-1}$$

The formula $\hat{\xi}_{j3}\hat{\gamma}_{j3}$ represents the availability for X_j. Here, $\hat{\xi}_{j3}$ and $\hat{\gamma}_{j3}$ are independent random variable.

$$E[\hat{\xi}_{j3}\hat{\gamma}_{j3}] = E[\hat{\xi}_{j3}] * E[\hat{\gamma}_{j3}] = \frac{\lambda_{oj}}{\mu_{oj}+\lambda_{oj}} * \frac{\lambda_{Rj}}{\mu_{Rj}+\lambda_{Rj}}$$

The formula $\hat{\xi}_{j4}\hat{\gamma}_{j4}$ represents the reliability for X_j. Assuming that $\hat{\xi}_{j4}$ and $\hat{\gamma}_{j4}$ are independent random variables. Then, $E[\hat{\xi}_{j4}\hat{\gamma}_{j4}]$ can be represented as the following formula:

$$\begin{aligned} E[\hat{\xi}_{j4}\hat{\gamma}_{j4}] &= E[\hat{\xi}_{j4}] * E[\hat{\gamma}_{j4}] \\ &= \int_0^\infty t dR_{Rj}(t) * \int_0^\infty t dR_{Oj}(t) \\ &= (1 - \frac{q_{10}^O(X_j)}{q_1^O(X_j)} * (1 + \frac{1}{(q_1^O(X_j))^2})) \\ &\quad * (1 - \frac{q_{10}^R(Y_j)}{q_1^R(Y_j)} * (1 + \frac{1}{(q_1^R(X_j))^2})) \end{aligned}$$

Through the above process, the uncertain stochastic linear programming problems are transformed into the certain equivalent types.

B. The Combination Policies

User preference policy is decided according to the user's subjective preference or cloud service provider's reputation. Data transfer policy is adapted to increase the data communication rate between activities. Those adjacent activities with more data transfer will be considered for assignment in the same cloud service provider. Sometimes, these policies are inconsistent, e.g., according to the user preference policy, two activities need to be assigned to different providers, while according to data transfer policy, the two activities should be assigned to the same or the closest cloud service providers which can improve the data communication rate. Therefore, the combined policy that considers user preference and data transfer is proposed.

Here, $Su = \{Su_s, Su_o\}$ represents user preference, where Su_s is user's subjective favorite site, and Su_o is the provider with better reputation. Suppose Sd represents the provider that can make data communication least between activities, and S' represents the final selection. The policy can be described as below.

1) When the data communication output can reach or exceed the given threshold, data transfer policy is considered firstly, and $S' = Sd$ is the selection;
2) Otherwise, if $S' = Su_s = Su_o$, S' is the selection;
3) And, else if the reputation of user's subjective preference provider Su_s is less than the given threshold, $S' = Su_o$ is the selection, else, $S' = Su_s$ is the selection.

C. The Optimization Algorithm

According to the probability distributions and Markov characteristics of the random variables, the uncertain stochastic model can be transformed into certain ILP(Integer Linear Programming) model. Therefore, the ILP approach can be used to get the solution. The ILP problem has been shown to be NP-complete [17]. Branch and bound algorithm can be adapted to find the optimal solution for it. The algorithm uses a search tree to get the solution and its complexity increases exponentially with the increasing problem size. Therefore, branch and bound algorithm is not suitable for large size problems.

The service processes decision problem for concurrent execution control fragments belongs to combinational optimization problem and can be mapped to a Multiple Choice Multiple Dimension Knapsack Problem (MMKP) as in [18]. The heuristic algorithms proposed in [19] may be useful to find feasible solution for MMKP. It has been proven to have the ability to find the suboptimal solutions in polynomial time. Furthermore, in the heuristic algorithm, it need be considered that concurrent execution branches can't select the same service to avoid service contention.

For the problem with only one integer variable and limited candidate services, the branch and bound algorithm can be suitable for finding the optimal scheme at runtime. In terms of the combinatorial optimization problem with two or more integer variables, the branch and bound algorithm has a high-time complexity and may not be suitable for systems with many services and dynamic service needs. Therefore, when QoS data are accurate and the problem size is small, the branch and bound algorithm can be use to solve this combinatorial optimization problem effectively, as proposed in [20,21]. For the large system with many service nodes and potential selections, this approach may not be acceptable for cloud service providers selection that need to make runtime decisions and have higher request to the computing efficiency of algorithm.

Therefore, mixed policy based on branch and bound algorithm and the modified heuristic algorithm is considered to find the solution for dynamic service processes decision problem.

The dynamic service processes decision algorithm are illustrated in detail as follows:

Algorithm 1:

INPUT ： CSWF-net W and its fragments set $F_W = \{F_{W_1}, F_{W_2}, \cdots, F_{W_m}\}$.

OUTPUT: The optimum cloud service providers $O' = \{O'_1, O'_2, \cdots, O'_m\}$ for each fragment.

Step 1: According to the control structure of the fragment F_{W_i} for $i = 1, 2, \cdots m$ and user requirement, the stochastic linear programming model P_{W_i} for F_{W_i} is defined, where $P_{W_i} \in \{P_1, P_2, P_3, P_4\}$.

Step 2: Then, the model P_{W_i} is transformed into expected value models EP_{W_i}.

Step 3: Expected value models EP_{W_i} is transformed into the certain equivalent type CP_{W_i} based on the distributions and Markov characteristics of its random variables.

Step 4: If $P_{W_i} = P_1$ or $P_{W_i} = P_3$, the branch and bound algorithm can be suitable for finding the optimal scheme at runtime for CP_{W_i} because these two models have only one integer variable and the candidate services is limit. If the branch and bound algorithm can't get feasible solutions, the modified heuristic algorithm is invoked.

Step 5: If $P_{W_i} = P_2$ or $P_{W_i} = P_4$, the QoS data are accurate, and the problem size is small, then branch and bound algorithm can be used to get the optimal solutions for CP_{W_i}. If the branch and bound algorithm can't get feasible solution, the modified heuristic algorithm is invoked.

Step 6: Else if $P_{W_i} = P_2$ or $P_{W_i} = P_4$, and the systems are large with many service nodes and potential selections, the modified heuristic algorithm is considered to get the suboptimal solutions for CP_{W_i}.

Step 7: If there are two or more optimization results $\{X_1^*, X_2^*, \cdots\}$ for CP_{W_i}, the combination policies of user preference and data transfer are considered. Then, the optimal solution X^* for CP_{W_i} is gotten.

Step 8: The cloud service providers Y^* that X^* belongs to is the optimal dynamic cloud service providers for the fragment F_{W_i}.

Step 9: In the same way, the cloud service providers for other fragments are selected. Finally, we can get the optimal cloud service providers $O' = \{O'_1, O'_2, \cdots, O'_m\}$ for each fragment.

V. EXPERIMENT AND ANALYSIS

Here, we take a product customization business for example to validate the proposed method. The customization business can be abstracted as a simplified business process as shown in Figure 2.

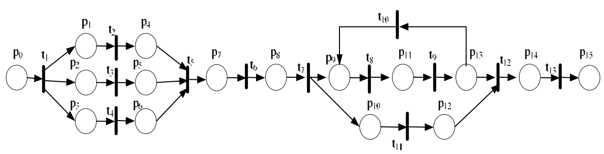

Figure 2. Simplified individual product customization process.

The process deals with customer requirement orders and coordinates the online services in cloud to finish production design, manufacturing, assembly, etc., collaboratively.

After the customers submit requirement order, the whole service process refers to order submission service, custom credit check-up service, designing service, manufacturing service, assembly service, etc. The individual product customization can be described as a constrained service workflow net. There are 8 roles $Rl = \{Rl_1, Rl_2, Rl_3, Rl_4, Rl_5, Rl_6, Rl_7, Rl_8\}$, and 5 cloud service providers $O = \{O_1, O_2, O_3, O_4, O_5\}$ participating process collaboration at current time T_c. The start time for each cloud service provider joining in the environment is $T_o^s = \{T_{o1}^s, T_{o2}^s, T_{o3}^s, T_{o4}^s, T_{o5}^s\}$, and 16 business service pools $Rp = \{Rp_1, Rp_2, \cdots, Rp_{16}\}$ are available. For Rp_i $(Rp_i \in Rp)$, the business services set can be denoted as $\{S_{i-1}, S_{i-2}, \cdots, S_{i-k_i}\}$, where k_i is the number of business services in business services pool Rp_i.

According to model fragmentation algorithm proposed in [3], the workflow model is separated into 13 fragments $F_W = \{F_{W_1}, F_{W_2}, F_{W_3}, F_{W_4}, \cdots, F_{W_{13}}\}$. Here, we take the concurrent execution fragments F_{W2}, F_{W3}, and F_{W4} (shown in Figure 3) for example to illustrate the optimum service processes stochastic decision approach proposed in this paper.

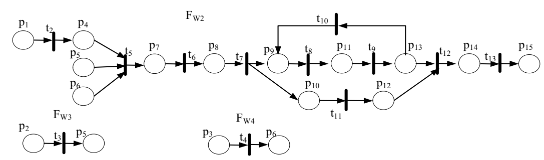

Figure 3. A parallel execution fragment for product customization process

Based on definition 2 and 3, the transition-role matrix *M*, the role-cloud service provider matrix *N*, and cloud service provider–business service matrix *L* are constructed.

$$M = \begin{bmatrix} 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 \\ 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0 \\ 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1 \end{bmatrix}^T$$

$$N = \begin{bmatrix} 1, 1, 1, 0, 1 \\ 1, 1, 1, 1, 1 \\ 1, 0, 0, 1, 1 \\ 0, 1, 1, 1, 1 \\ 1, 1, 1, 1, 0 \\ 1, 1, 1, 0, 0 \\ 1, 0, 1, 1, 1 \\ 1, 1, 1, 1, 1 \end{bmatrix}^T$$

$$L = \begin{bmatrix} 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1 \\ 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1 \\ 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1 \\ 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0 \\ 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1 \end{bmatrix}^T$$

Obviously, role Rl_2 is responsible for the execution of F_{W2} and F_{W3}, and Rl_3 corresponds to F_{W4}. According to definition 2, 3, 4 and 5, we can acquire the transition- cloud service provider matrix $B(F_{W2}, F_{W3}, F_{W4})$ and the transition-business service pools matrix $A''(F_{W2}, F_{W3}, F_{W4})$ for F_{W2}, F_{W3}, and F_{W4}.

$$B(F_{W2}, F_{W3}, F_{W4}) = B(T_2, T_3, T_4) = \begin{bmatrix} 1, 1, 1, 0, 1 \\ 1, 1, 1, 0, 1 \\ 0, 1, 1, 1, 0 \end{bmatrix}^T$$

$$A''(F_{W2},F_{W3},F_{W4}) = A''(T_2,T_3,T_4) = \begin{bmatrix} 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1 \\ 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1 \\ 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0 \end{bmatrix}^T$$

For F_{W2} and F_{W3}, the candidate services set can be represented as:

$$\{S_{1-1},\cdots,S_{1-k_1},S_{4-1},\cdots,S_{4-k_4},S_{5-1},\cdots,S_{5-k_5},S_{9-1},\cdots,S_{9-k_9},S_{16-1},\cdots,S_{16-k_{16}}\}$$

And the candidate services set for F_{W4} is denoted as:

$$\{S_{7-1},\cdots,S_{7-k_7},S_{8-1},\cdots,S_{8-k_8},S_{13-1},\cdots,S_{13-k_{13}}\}$$

We assume that the customers have multiple QoS objectives. Service processes stochastic decision problem under multiple QoS objectives is considered. Assuming that the weights for the four metrics are $W = \{0.25, 0.20, 0.35, 0.20\}$, and the QoS thresholds are $Q_t = 2.8$, $Q_c = 65$, $Q_a = 0.91$, and $Q_r = 0.93$. $T_o^s(X_j)$ is the start time that the cloud service provider joins in service processes, and $T_o^c(X_j)$ is the current time.

The QoS criteria are normalized by equation (1a) and (1b). The fragments F_{W2}, F_{W3} and F_{W4} are concurrent execution fragments and can be modeled as the problem P4. Then P4 can be transformed into EVM. Based on equations 2-4, we can get the equivalent certain ILP problem of EVM. The ILP problem can be solved by the branch-and-bound algorithm effectively since its size is small. When $X = \{S_{16-1}, S_{1-2}, S_{13-2}\}$, $F(X)$ obtains the maximum value 0.897. Therefore, the optimum cloud service providers for the three fragments $\{F_{W2},F_{W3},F_{W4}\}$ are $\{O_5,O_1,O_4\}$.

Accordingly, the cloud service providers for other fragments can be gotten based on the algorithm 1 by the above process.

VI. Conclusions and Future Work

Based on the approach proposed in this paper, the optimum service process scheme can be configured at runtime to support the execution of virtual business effectively.

Compared with the current methods in which the cloud service providers are selected on the fly, the proposed stochastic service processes configuration and optimization approach improves the execution performance of virtual business greatly.

The further need is to consider the set of candidate business services which satisfy the end user preferences for the whole ecosystem. The qualitative QoS criteria influencing service processes execution also need to be considered based on semantic technique. Moreover, the design and development of a suite of software tools to support the service processes optimization in service-oriented business ecosystem is also important for the spread and application of the approach.

Acknowledgements

This work is supported by National Key Technology R&D Program under Grant No. 2011BAF10B01.

References

[1] Sufen Li,Yushun Fan and Xitong Li. A trust-based approach to selection of business services , Computer Integrated Manufacturing, 24(7), 2011: 769-784

[2] H. Motahari-Nezhad, B. Stephenson, and S. Singhal, Outsourcing Business to Cloud Computing Services: Opportunities and Challenges, Available at www. hpl.hp.com/techreports/2009/HPL-2009-23.html, 2009

[3] W. Tan, Y. S. Fan, Dynamic workflow model fragmentation for distributed execution, Computers in Industry 58(5) ,2007:381-391.

[4] G. B. Chafle, Orchestrating composite Web services under data flow constraints, Proc. IEEE International Conference on Web Services (ICWS 2005), vol. 1, 2005, pp. 211-218.

[5] B. X. Liu, Y. F. Wang, A role-based approach for decentralized dynamic service composition, Journal of Software 16(11), 2005: 1859-1867 (in Chinese).

[6] W. Tan, Y. S. Fan, Model fragmentation for distributed workflow execution: A Petri net approach, Proc. 5th International School and Symposium on Advanced Distributed Systems, ISSADS, vol. 3563, 2005, pp. 207–214.

[7] Menascé, D.A., Composing Web Services: A QoS View. IEEE Internet Computing, 8(6), 2004: 88-90.

[8] Li, L., Wei, J., and Huang, T, High Performance Approach for Multi-QoS Constrained Web Services Selection. Lecture Notes in Computer Science. Springer-Verlag Berlin Heidelberg, 4749, 2007: 283-294.

[9] S. L. Liu, Y. X. Liu, F. Zhang, G. F. Tang, and N. Jing, "Dynamic web services selection algorithm with QoS global optimal in web services composition", Journal of Software, vol. 18, 2007: 646-656,.

[10] Huang, C. W. Lan and S. Yang, "An optimal QoS-based Web service selection scheme", INFORMATION SCIENCES, vol. 179, 2009: 3309-3322

[11] P. Doshi, R. Goodwin, R. Akkiraju, and K. Verma. Dynamic workflow composition using markov decision processes. Journal of Web Services Research (JWSR), 2(1), 2005:1-17

[12] W. M. P. Van Der Aalst, The application of petri nets to workflow management, The Journal of Circuits, Systems and Computers, 8(1) (1998) pp. 21–66.

[13] W. M. P. van der Aalst and Arthur H. M. ter Hofstede, Verification of workflow task structures: a petri-net-based approach, Information Systems, 25(1) (2000) 43-69.

[14] C. L. Hwang, K. Yoon, Multiple criteria decision making – methods and application, a state-of-art survey, Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, New York, 1981.

[15] S. Ran, A model for Web services discovery with QoS, SIGecom Exchanges, 4(1) (2003) 1-10.

[16] B. Liu, Theory and practice of uncertain programming, Physica-Verlag Heidelberg, 2003.

[17] T. Yu , Y. Zhang, K.J. Lin, Efficient algorithms for Web services selection wit h end to end QoS constraints, ACM Transactions on Web, 1(1) (2007) 1-26.

[18] T. Yu , K. J. Lin, Service Selection Algorithms for Composing Complex Services with Multiple QoS Constraints, Proc. International Conference on Service-Oriented Computing (ICSOC 05), vol. 3826, 2005, pp. 130-143.

[19] S.Y. Ye, J. Wei, L. Li, etc, Service-correlation aware service selection for composite service, Chinese Journal of Computers, 31(8) (2008) 1383-1397.

[20] L. ZENG, B. BENATALLAH, A. NGU, etc, Quality-aware middleware for Web service composition, IEEE Transactions on Software Engineering, 30(5) (2004) 311–327.

[21] X.W. Chen, H.F. Ou, and Q.P. Zhao, A distributed workflow management model for grid middleware, International Journal of Parallel, Emergent and Distributed Systems, 23(2) (2008) 107-120