Aug. 2014 Transactions of Nanjing University of Aeronautics and Astronautics Vol. 31 No. 4

Semantic Social Service Organization Mechanism in Cyber Physical System

Zhang Bo（张波）*，*Pan Xiaosheng*（潘晓声），*Pan Jianguo*（潘建国）

College of Information，Mechanical and Electrical Engineering，Shanghai Normal University，Shanghai，
200234，P. R. China

（Received 19 April 2013；revised 8 July 2013；accepted 11 July 2013）

Abstract：Cyber physical system (CPS) provides more powerful service by cyber and physical features through the wireless communication. As a kind of social organized network system，a fundamental question of CPS is to achieve service self-organization with its nodes autonomously working in both physical and cyber environments. To solve the problem，the social nature of nodes in CPS is firstly addressed，and then a formal social semantic descriptions is presented for physical environment，node service and task in order to make the nodes communicate automatically and physical environment sensibly. Further，the Horn clause is introduced to represent the reasoning rules of service organizing. Based on the match function，which is defined for measurement between semantics，the semantic aware measurement is presented to evaluate whether environment around a node can satisfy the task requirement or not. Moreover，the service capacity evaluation method for nodes is addressed to find out the competent service from both cyber and physical features of nodes. According to aforementioned two measurements，the task semantic decomposition algorithm and the organizing matrix are defined and the service self-organizing mechanism for CPS is proposed. Finally，examinations are given to further verify the efficiency and feasibility of the proposed mechanism.

Key words：cyber physical system (CPS)；self-organizing mechanism；semantic；environment-aware measurement；service capacity evaluation

CLC number：TP391 **Document code**：A **Article ID**：1005-1120(2014)04-0452-11

1 Introduction

Cyber physical system (CPS) is a kind of mobile sensor network system which combines the virtual computational world and physical world into united system in order to provide more powerful service capacities and qualities to human beings[1]. As a kind of environment-sensitive and dynamically organized system，CPS should be not only adaptive to physical and cyber environments，but with self-organizing capacity according to application needs[2]. Therefore，a unique challenge of CPS is how to organize services best matched the collaborative applications from the nodes in the CPS's mobile sensor network.

One important difference between CPS network and the existing internet technology is that the communication transmitted via CPS network is usually the measurement on the physical world and therefore subject to certain constraints due to physical environments[3]. Many examples of CPS，such as intelligent traffic control，smart buildings，and sensor devices，show that the performance of CPS is closely bound up with the organizing of the node devices which provide various services for the whole system.

Existing researches in CPS mainly focused on architecture[4]，middleware designing[5]，system

---

Foundation items：Supported by the National Natural Science Foundation of China (61103069，71171148)；the National High-Tech Research and Development Plan of China ("863" Plan)(2012BAD35B01)；the Innovation Program of Shanghai Municipal Education Commission (13YZ052)；the Shanghai Committee of Science and Technology (11DZ1501703，11dz12106001)；the Program of Shanghai Normal University (DXL125，DCL201302).

* **Corresponding author**：Zhang Bo，Associate Professor，E-mail：sh.zhangbo@gmail.com.

control[6], system security[7], QoS[8] or real-time data management[9]. However, as service provider, it is critical to organize a competent service group for CPS autonomously, which receives limited attention in study. Efforts on node organizing are usually based on criteria of cost minimization or utility maximization[10,11]. In our view, such an environment sensitive self-organizing mechanism of CPS should be based on cognizable service capacity, qualified service measurement, and self-communication between sensor nodes of CPS.

In addition, there is another significant feature, social relation, which is easy to be ignored. Social online service, such as social networks and microblog, is popular applications for users[2,12,13]. In real world, people also tend to cooperate with familiar persons because of the creditable relationship. Likewise, social relations, such as relationships or communities, are also important for nodes cooperation organizing in CPS. We can organize nodes in CPS through their past relationships and evaluate whether they can complete the task collaboratively or not.

We address an environment aware service self-organizing mechanism for CPS. Our main contributions here are: (1) Formal semantic descriptions for CPS node, physical environment, task and social relation are presented so that CPS can be understandable and communicatable based on the formal semantics; (2) The reasoning rule of service organizing for CPS are represented based on Horn clause; (3) Self-organization algorithms of CPS service, including environment aware measurement and service capacity evaluation selection, are proposed based on semantic decomposition, and organizing matrix are defined for CPS. Finally, we propose the service self-organizing mechanism for CPS based on aforementioned method.

2 Semantic Descriptions of CPS

Formal semantic description should be employed to enable nodes of CPS to recognize the meanings of both physical and cyber aspects and make self-communication between nodes possible. In our views, formal semantic for CPS service organizing must be defined from the following aspects: capacity of node, physical environment, requirement of task and social relationship.

In the section, we define the formal semantic descriptions for our mechanism: CPS node, physical environment, and application task.

2.1 Capacity of node

CPS node is a service provider through its physical device and cyber software. Thus, we describe capacity of CPS node in terms of physical and cyber features.

Semantic of physical capacity can be defined as follows:

Definition 1 Physical capacity semantic can be described as $Ps=(id, I|O, \Im\_\mathrm{real}, \Im\_\mathrm{mobile}, p\_\mathrm{value}$. Here, id denotes the identity code of device in CPS, $I|O$ the input port parameters and output port parameters of devices, $\Im\_\mathrm{real}$ the real-time environment around the node, $\Im\_\mathrm{mobile}$ a set of environment as $(\Im_1^M, \Im_2^M, \cdots)$, which denotes the potential environments that device can move, p_value the physical capacity performance value of node.

In definition 1, physical capacity performance value is evaluated from three aspects: fault tolerance, adaptability and stability.

Let a node d_i which has totally success rate of work in the past be $\mathrm{suc}(d_i)$, and the fault rate in the past be $\mathrm{fault}(d_i)$. At the same time, let the rate of node′s recovering from the faults be $\mathrm{recover}(d_i)$. The node′s capacity of fault tolerance can be calculated as

$$cp\_t(d_i)=\mathrm{suc}(d_i)^{1-\mathrm{fault}(d_i)}\times \mathrm{recover}(d_i)^{\mathrm{fault}(d_i)} \tag{1}$$

In Eq. (1), we can see that the lower value of $\mathrm{fault}(d_i)$ is, the higher value of capacity of fault tolerance $cp\_t(d_i)$ is.

Suppose there are a node d_i with an environment list $E=(e_1, e_2, \cdots)$ which indicates the detail environments d_i working around and $\mathrm{ratio}(e_k)$ which denotes the successful ratios of each environment e_k ($\sum \mathrm{ratio}(e_k)=1$). Assume that d_i

works n times in environment list $E=(e_1, e_2, \cdots)$ in the past. Then, for a task environment list $E\_\text{task}=(w_1, w_2, w_3, \cdots)$ physical capacity value of adaptability of d_i can be calculated as

$$cp\_a(d_i)=\frac{\sum\limits_{\text{match}(e_k, E\_\text{task})}\text{ratio}(e_k)}{|E\_\text{task}|}\times\beta^{\frac{1}{n}} \tag{2}$$

In Eq. (2), function $\text{match}(e_k, E\_\text{task})$ is a match expression and it indicates that there be an environment $w_l \in E\_\text{task}$ which satisfies the condition $(\exists e_k \in E)\rightarrow(e_k=w_l)$. Function $|E\_\text{task}|$ indicates the total number of list E_task. $\beta\in[0,1]$ is a regulation parameter for calculation and is an empirical value which is given in advance based on the experiences of examinations. We can see that the higher value of in Eq. (2) implies the value of $cp\_a(d_i)$ will be higher.

Let node d_i has stability performances as follows: $|\text{Time}_{\text{exist}(d_i)}|$ denotes the total length of d_i existing time and $|\text{Time}_{\text{stable}(d_i)}|$ indicates the total stable working time length of d_i. Assume that node d_i failures n times from CPS in the past. Then, physical capacity value of stability can be calculated as

$$cp\_s(d_i)=\frac{|\text{Time}_{\text{stable}(d_i)}|}{|\text{Time}_{\text{exist}(d_i)}|}\times\beta^{(1-\frac{1}{n})} \tag{3}$$

where $\beta\in[0,1]$ is the parameter which is given the same as in Eq. (2).

Therefore, physical capacity value can be calculated as

$$p\_\text{value}(d_i)=\frac{1}{3}\times[cp\_t(d_i)-cp\_a(d_i)-cp\_s(d_i)] \tag{4}$$

Semantic of physical device describes the physical information about nodes in CPS, including interface, mobility and capacity measurement, so that nodes in CPS can recognize each other and communicate autonomously. All elements of Ps are some of the parts which influence the service quality in physical degree directly.

Semantic of cyber capacity of service aims to provide the all software descriptions of the CPS nodes. CPS depends on cyber components to provide the service solutions to meet user's needs. In order to achieve the self-organizing requirement, CPS should know the cyber capacities of nodes and integrate them into a whole for solving complex tasks. Thus, we need to make the cyber components apprehensible and readable for CPS.

Definition 2 Cyber capacity semantic of service can be denoted as $Cs=(\text{class}, \text{in}|\text{out}, \text{timeliness}, \text{price}, c\_\text{value})$. Here, class is the class name of the cyber service, parameter pair in |out denotes the input and the output data formats and values of the service, timeliness indicates a period which service must spend for completing a task, price denotes the average price of the service respectively, $c_$value is cyber capacity index which demonstrates the performance of service software aspects based on the past working.

In definition 2, cyber capacity index value is evaluated from two aspects: timeliness and past judgment.

For each node d_i, it has a time list $\text{Time}=(t\_d_i^1, t\_d_i^2, t\_d_i^3, \cdots)$ $(t\_d_i^1<t\_d_i^2<t\_d_i^3<\cdots)$ which indicates time costs in the past for the same task and ratio $(t-d_i^k)$ denotes the occurrence ratio of time cost $t-d_i^k(\Sigma\text{ratio}(t-d_i^k)=1)$. Assume that d_i participates the task n times in the past. Then, for an anticipant working time, dt, capacity value of timeliness can be calculated as

$$cc\_t(d_i)=\sum_{t-d_i^k\leqslant dt}\text{ratio}(t-d_i^k)\times\beta^{\frac{1}{n}} \tag{5}$$

where $\beta\in[0,1]$ is the parameter which is given the same as in Eq. (2).

Let node d_i provide service n times in the past. Each time of service generates a judgment value $\text{jud}(d_i)$ $(\text{jud}(d_i)\in[0,1]$. Assume that there are m times of malicious judgments to d_i. Then, the capacity value of past judgment can be calculated as

$$cc\_j(d_i)=\frac{\sum\text{jud}(d_i)}{n}\times\left(\frac{n-m}{n}\right)^{\frac{1}{n-m}} \tag{6}$$

Therefore, cyber capacity value, $c_$value, can be calculated as

$$c\_\text{value}(d_i)=\frac{1}{2}\times[cc\_t(d_i)+cc\_j(d_i)] \tag{7}$$

Based on definitions 1,2, each node in CPS can be described as node=(Ps, Cs).

2.2 Physical environment

Physical environment is an objective concept which describes the elements around the CPS node, including location, time or status, etc. We define semantic of physical environment as follows:

Definition 3 Physical environment can be described as $\mathfrak{J}=(L, P, S, C)$, where L, P, S and C represent the physical information sets of location, time, status and constraint, respectively.

Here, $L=(loc_1, loc_2, \cdots)$ is space set to describe the physical location information of the objects in CPS. $P=(p_1, p_2, \cdots)$ is set to describe the planning time information. $S=(s_1, s_2, \cdots)$ is the set of status descriptions for nodes in CPS. $C=\{c_i(x_j) \mid x_j \in L \vee P \vee S\}$ is constraint which can identify the physical environment semantic elements.

2.3 Task semantic

Task denotes application needs from users. It gives all pre-conditions as a criterion for CPS nodes to measure whether they can achieve the task's goals or not. Therefore, in definition 3, we describe the task semantic from both cyber and the physical aspects as follows:

Definition 4 Task semantic is a set as $T=(T_1, T_2, \cdots)$, where sub-task T_i can be defined as $T_i=(T\text{Class}_i, \text{Status}_i, \text{Goal}_i, \text{Cost}_i, \text{Envir}_i, \text{Cons}_i)$. Here, $T\text{Class}_i$ denotes the class of task, Status_i the set of original facts which are input data given in advance, Goal_i the set of task's anticipative goals, Cost_i, Envir_i and Cons_i denote the requirements of cost, physical environment and constraints for the task, respectively.

As for environment requirements, Envir_i is described as semantic format in definition 3. And semantic of constraints, Cons_i, indicates the limitations for nodes which tends to accept the task except environment constraints. For example, there is a sub-task semantic as

T_i=(send_data,{node_1},{node_3},{10},
{(community_1), (13: 00 — 14: 00 AM), (free),(before14:00)},($\text{node} \in \text{community}_1$))

We can see send_data denotes the type of task is sending data, node_1 and node_3 point out that the task is to send data from original node node_1 to goal node node_3. At the same time, location of node_1 is community_1, anticipant working period is 13:00—14:00 AM, node_1 is in free status and the work must be completed before 14:00 AM. In addition, the task assigns that it must be executed by node which belongs to community_1.

2.4 Social semantic

According to the organization method of social network, each node has the information which records the node's communities and its relationships or friends. These social items of node can be acquired from node personal information or profiles. In this paper, social semantic of node in CPS describes the detail relationships among nodes and communities where nodes locate in. We define social semantic of node as follows:

Definition 5 Social semantic of node is a set as S=(Community, Relationship). Here, Community=(c_1, c_2, $\cdots$) denotes the communities which nodes locate in and Relationship=($r_1(d_i)$, $r_2(d_j)$, $\cdots$) denotes the relationships between nodes.

Social semantic indicates node's identification in CPS. Service organizer can obtain the information of working community and relationship through social semantic and then takes appropriate measures for organizing.

3 Rule Based Semantic Reasoning for Service Organizing

3.1 Horn clause

Horn clause is a clause (a disjunction of literals) with at most one positive literal[14]. It plays a basic role in logic programming and is important for constructive logic. Due to Horn clause contains at most one positive literal, it is widely used in knowledge reasoning and rule representation[15]. In this paper, we introduce Horn clause in order to realize the logical condition reasoning for service self-organizing.

Definition 6 Horn clause is a clause which has one positive literal at most. Horn clause can be written in form as follows

$$P_1 \wedge P_2 \wedge \cdots \wedge P_n \rightarrow Q \tag{8}$$

where P_i and Q are propositions, and $\wedge$, $\rightarrow$ are logic connectors. Eq. (8) means that the literal Q can be realized while all the propositions P_i can be satisfied.

For reasoning, Horn clause can be expressed three formats as follows:

(1) A Horn clause with both positive literals P_i and Q is called a rule clause, which is in form of $P_1 \wedge P_2 \wedge \cdots \wedge P_n \rightarrow Q$.

(2) A Horn clause with no positive literals Q is called a goal clause, which is in form of $P_1 \wedge P_2 \wedge \cdots \wedge P_n \rightarrow$.

(3) A Horn clause with no precondition literals P_i is called a fact, which is in form of $\rightarrow Q$.

Based on definition 6, we can achieve self-organizing for service selection in formal representing and reasoning through Horn clause. For example, goal solving by node can be represented as $\mathrm{in}_1 \wedge \mathrm{in}_2 \wedge \cdots \wedge \mathrm{in}_n \rightarrow \mathrm{out}$, and the status of task semantic can be represented as $\rightarrow \mathrm{status}_i$.

3.2 Rule based environment measurement

We propose several constraints for environment semantic in order to make environment measurement feasible.

Constraint 1 Each environment can be described as a fact in form of Horn clause $\rightarrow \Im$.

Constraint 1 shows that all elements of environment semantic, including L, P, S and C, are positive literals.

Constraint 2 For task semantic, requirement of environment can be described as a rule in form of Horn clause as $\Im \rightarrow T_i.\ \mathrm{Envir}_i$.

Nodes of CPS work with physical limitations which we define as environment in the paper. Environment aware measurement aims to evaluate whether the real-time environment around a node can satisfy the task environment requirement or not. Firstly, we propose a match function for our measurement.

Definition 7 Let X be a set of facts and y be a requirement. We define a match function $X \mid\rightarrow y$ while there is a the set of fact X which can meet the requirement y in form of Horn clause as follows

$$X \mid\rightarrow y \equiv x_1 \wedge x_2 \wedge \cdots \wedge x_n \rightarrow y \tag{9}$$

where, $x_1 \in X, \cdots, x_n \in X$.

Then, we propose the semantics of match function in physical environment as

$$\begin{cases} L \mid\rightarrow loc_1 \equiv \exists\, loc_2 \in L \wedge loc_2 = loc_1 \\ P \mid\rightarrow p_1 \equiv \exists\, p_2 \in P \wedge p_1.\mathrm{st} \leqslant p_2. \\ \quad \mathrm{st} \wedge p_2.\mathrm{et} \leqslant p_1.\mathrm{et} \\ S \mid\rightarrow s_1 \equiv \exists\, s_2 \in S \wedge s_2 = s_1 \\ C \mid\rightarrow c_1 \equiv \neg\ \exists\, c_2 \in C \wedge c_1 = \neg\ c_2 \end{cases} \tag{10}$$

where $p_1.\mathrm{st}$ denotes the starting time and $p_1.\mathrm{et}$ the ending time of parameter p_1.

Based on definition 7, we can define environment measurement as a group of rule Horn clause as

$$\begin{cases} \Im.L.loc_1 \wedge \Im.L.loc_2 \wedge \cdots \rightarrow T_i.\mathrm{Envir}.L \\ \Im.P.p_1 \wedge \Im.P.p_2 \wedge \cdots \rightarrow T_i.\mathrm{Envir}.P \\ \Im.S.s_1 \wedge \Im.S.s_2 \wedge \cdots \rightarrow T_i.\mathrm{Envir}.S \\ \Im.C.c_1 \wedge \Im.C.c_2 \wedge \cdots \rightarrow T_i.\mathrm{Envir}.C \end{cases} \tag{11}$$

Let an environment requirement of task be T_i. Envir. For each $\Im$, environment measurement can be defined as following algorithms.

Algorithm 1 Environment measurement

Step1

$$N_1 \leftarrow |\ T_i.\mathrm{Envir}.L\ | + |\ T_i.\mathrm{Envir}.P\ | + |\ T_i.\mathrm{Envir}.S\ | + |\ T_i.\mathrm{Envir}.C\ |$$

Step2 For each requirement x_j, which has relationship with $x_j \in T_i.\mathrm{Envir}.L \cup T_i.\mathrm{Envir}.P \cup T_i.\mathrm{Envir}.S \cup T_i.\mathrm{Envir}.C$, $\Im$ utilizes match function to find out whether there is a set of location facts $\Im.L$ with $\Im \mid\rightarrow x_j$.

Step3 If there is x_j which can meets function $\Im \mid\rightarrow x_j$, then $N_1 \leftarrow N_1 - 1$.

Step4 Repeat steps 2 and 3 until all requirements x_j have been measured by match function.

Step5 If $N_1 = 0$, the requirement T_i. Envir is satisfied by $\Im$ and $\Im \rightarrow T_i$. Envir. And else, there is x_j which can not be satisfied by match function.

For example, suppose there are environment semantic $\Im_1 = (loc_1, [20{:}00\text{—}22{:}00], (s_1,\ s_2,$

s_3), $\varnothing$), and environment requirement , then environment aware measurement can output the rule $\Im \rightarrow T_i.\text{Envir}$.

3.3 Rule based capacity evaluation

For feasible service selection, we also propose several constraints for semantic in order to make the descriptions more clearly understandable for CPS node.

Constraint 3 For each task semantic, there is a constraint as

$$T_i.\text{Goal} \neq \varnothing \wedge T_i.\ \text{Status} \neq \varnothing$$

This constraint means task semantic will be invalid while it does not specify its goals and preconditions.

Constraint 4 Goal constraint denotes that for a set of task $T=(T_1, T_2, \cdots)$, there is a constraint as

$$\forall T_i \in T \wedge \forall T_j \in T \rightarrow T_i.\text{Goal} \cap T_j.\text{Goal}_j = \varnothing$$

That means all sub-tasks of T cannot have same goals.

Constraint 5 For a node, it has an $I|O$ constraint of cyber service and physical device as

$$\begin{cases} \text{node}.Ps.i \wedge \text{node}.Cs.\text{in} \wedge (\text{node}.Ps.i \mid\rightarrow \\ \quad \text{node}.Cs.\text{in}) \rightarrow \\ \text{node}.Ps.O \wedge \text{node}.Cs.\text{out} \wedge (\text{node}.Ps.O \mid\rightarrow \\ \quad \text{node}.Cs.\text{out}) \rightarrow \end{cases} \tag{12}$$

Constraint 5 points out that all node services should have a concordant input or output ports for service providing. Here, $\text{node}.Ps.i \mid\rightarrow \text{node}.Cs.\text{in}$ means that the physical device input port can match the cyber service input port. Likewise, $\text{node}.Ps.O \mid\rightarrow \text{node}.Cs.\text{out}$ denotes similar meanings.

We suppose that all nodes of CPS are the candidate service providers to solve complicated task, and nodes of CPS have various capacities with own prices. The net result of the above consideration is that CPS needs to find a feasible solution to evaluate competent services from nodes and reach a self-organized, temporary, and efficient service composition according to the task.

Capacity evaluation is an effective solution for CPS service selection. CPS can decide whether a node's service should be selected or not depending on its performance. To evaluate the performance of a candidate service, CPS should match the capacities of node service with tasks.

For a sub-task T_i, node in CPS node_j can perform the task T_i while the following group of rule Horn clause can be satisfied

$$\begin{cases} T_i.\text{Status}_i \rightarrow \text{node}_j.Ps.I \\ T_i.\text{Status}_i \rightarrow \text{node}_j.Cs.\text{in} \\ \text{node}_j.Ps.O \wedge \text{node}_j.Cs.\text{out} \rightarrow T_i.\text{Goal}_i \\ \text{node}_j.Cs.\text{price} \rightarrow T_i.\text{Cost}_i \end{cases} \tag{13}$$

We propose the semantics of match function in service evaluation as

$$\begin{cases} T_i.\text{Status} \mid\rightarrow \text{node}_j.Ps.i \equiv \text{node}_j.Ps.i_j \in T_i.\text{Status} \\ \text{node}_j.Ps.O \mid\rightarrow T_i.\text{goal} \equiv T_i.\text{goal} \in \text{node}_j.Ps.O \\ T_i.\text{Status}_i \mid\rightarrow \text{node}_j.Cs.\text{in} \equiv \text{node}_j.Cs.\text{in} \in T_i.\text{Status}_i \\ \text{node}_j.Cs.\text{out} \mid\rightarrow T_i.\text{goal} \equiv T_i.\text{goal} \in \text{node}_j.Cs.\text{out} \\ \text{node}_j.Cs.\text{price} \mid\rightarrow T_i.\text{cost} \equiv \text{node}_j.Cs.\text{price} \in T_i.\text{cost} \end{cases} \tag{14}$$

Suppose there is a sub-task T_i. For all nodes of CPS, service capacity evaluation consists of 6 steps as follows:

Algorithm 2 Service capacity evaluation

Step 1 $N_2 \leftarrow T_i.\text{goal}$.

Step 2 For each $\text{goal} \in T_i.\text{Goal}$, match the goal with function as $\text{node}_j.Ps.O \wedge \text{node}_j.Cs.\text{out} \mid\rightarrow T_i.\text{goal}$. If the function can be satisfied, $N_2 \leftarrow N_2 - 1$.

Step 3 Repeat steps 2 and 3 until all goals are evaluated by match function.

Step 4 If $N_2 = 0$, go to step 5. Otherwise, algorithm finishes. Quit the service evaluation and $\text{evaluate}(\text{node}_j) = 0$.

Step 5 For each $\text{node}_j.Ps.i$ and $\text{node}_j.Cs.\text{in}$, measure whether the input ports can be satisfied by the facts of $T_i.\text{Status}$. If input ports cannot be satisfied by facts, quit this algorithm and $\text{evaluate}(\text{node}_j) = 0$. Otherwise, go to the next step.

Step 6 Evaluate the cost requirement of T_i. If match function $\text{node}_j.Cs.\text{price} \mid\rightarrow T_i.\text{cost}$ is satisfied, the Horn clause $\text{node}_j \rightarrow T_i$ is valid and $\text{evaluate}(\text{node}_j) = c\_\text{value}(d_j)$. Otherwise, $\text{evaluate}(\text{node}_j) = \frac{1}{2} \times c\_\text{value}(d_j)$.

Here, we propose an example to illustrate our algorithm. For $T_2.\mathrm{Status}_2=(s_1, s_2, s_3)$, $T_2.\mathrm{Goal}_2=t_1\,|\,g_1$, $T_2.\mathrm{Cost}_2=[0,10]$, there be $\mathrm{node}_1=[I=(s_1)\,|\,O=(t_1),\ \mathrm{in}=(s_3)\,|\,\mathrm{out}=(g_1),\ \mathrm{price}=13,0.9]$, $\mathrm{node}_2=[I=(s_1)\,|\,O=(t_1),\ \mathrm{in}=(s_3)\,|\,\mathrm{out}=(g_1),\ \mathrm{price}=5,1.0]$ and $\mathrm{node}_2=[I=(s_1,s_4)\,|\,O=(t_1),\ \mathrm{in}=(s_3)\,|\,\mathrm{out}=(g_1),\ \mathrm{price}=14,1.0]$. Evaluation value of the above three nodes can be evaluated as 0.45, 1 and 0 respectively by algorithm 2.

4 Service Self-Organizing for CPS

4.1 Task semantic decomposition

Through capacity evaluation, we can evaluate whether a task can be solved by CPS nodes. However, it is impossible to find single node which can provide scenario for every task. In some case, a node can just solve a part of goals of a complicated task. In this paper, we address a task semantic decomposition method in order to decompose task into two sub-tasks according to their goals and make the task solving feasible.

For a task T_i, there is a set of goals $T_i.\mathrm{Goal}'\subset T_i.\mathrm{Goal}$ which cannot be solved by a node. We can decompose it into two task semantics, T_i^{d1} and T_i^{d2}, as in definition 8.

Definition 8 Task semantic decomposition can be defined as

$$\begin{cases} T_i^{d1}=[T_i.T\mathrm{Class},(T_i.\mathrm{Status}-T_i.\mathrm{Status}'), \\ \quad (T_i.\mathrm{Goal}-T_i.\mathrm{Goal}'),(T_i.\mathrm{Cost}-T_i.\mathrm{Cost}'), \\ \quad (T_i.\mathrm{Envir}-T_i.\mathrm{Envir}')] \\ T_i^{d2}=[T_i.T\mathrm{Class},T_i.\mathrm{Status}',T_i.\mathrm{Goal}',T_i.\mathrm{Cost}', \\ \quad T_i.\mathrm{Envir}'] \end{cases} \tag{15}$$

where $T_i.\mathrm{Status}'\subset T_i.\mathrm{Status}$, $T_i.\mathrm{Cost}'<T_i.\mathrm{Cost}$ and $T_i.\mathrm{Envir}'\subset T_i.\mathrm{Envir}$ are corresponding requirements of status, cost and environment of $T_i.\mathrm{Goal}'$. Eq. (15) means that complex task can be decomposed into two sub-tasks, T_i^{d1} and T_i^{d2}, which satisfy the condition $T_i^{d1}\cup T_i^{d2}=T_i$.

In our consideration, a task, which can not be solved by any single node, should be decomposed into sub-tasks with the minimum number for reducing computing complexity. Therefore, we propose the algorithm of task decomposition for CPS.

Algorithm 3 Task semantic decomposition

Step 1 There are a task T_i, and a set of nodes Node. For each $\mathrm{node}_j\in\mathrm{node}$, CPS builds up schema $\mathbf{R}_j$ to record the goals which can be solved by node_j.

Step 2 For each $\mathrm{node}_j\in\mathrm{node}$, CPS measures $T_i.\mathrm{goal}_k$ with matching function in Eq. (14).

Step 3 If the rule of $\mathrm{node}_j.D.O\wedge\mathrm{node}_j.N.\mathrm{out}\rightarrow T_i.\mathrm{goal}_k$ is true, $\mathbf{R}_j\leftarrow T_i.\mathrm{goal}_k$.

Step 4 Repeat steps 2 and 3 until all nodes get $\mathbf{R}_j$.

Step 5 CPS calculates the set compositions with constraint as $(\bigcup\mathbf{R}_j=T_i.\mathrm{Goal})\wedge(\bigcap\mathbf{R}_j=\varnothing)$.

Step 6 CPS gets the $\mathbf{R}_j$ composition with the minimum sub-task number. The task T_i is decomposed as in definition 8.

For example, there is a task T_3 can not be solved by any single node in CPS. The decomposition is shown in Table 1.

Table 1 Example of task semantic decomposition

T_3.Goal	g_1	g_2	g_3	g_4
node_4	×	×	√	×
node_5	√	×	×	√
node_6	√	√	×	√

Symbol "×" means that the goal can not be achieved by corresponding node while "√" means that the goal can be achieved.

From Table 1, we can decompose T_3 into two sub-tasks as

$$\begin{cases} T_3^{d1}=(T_3.T\mathrm{Class},T_3.\mathrm{Status}^{g_3},(g_3), \\ \quad T_3.\mathrm{Cost}^{g_3},T_3.\mathrm{Envir}^{g_3}) \\ T_3^{d2}=(T_3.T\mathrm{Class},T_3.\mathrm{Status}^{g_1,g_2,g_4},(g_1,g_2,g_4), \\ \quad T_3.\mathrm{Cost}^{g_1,g_2,g_4},T_3.\mathrm{Envir}^{g_1,g_2,g_4}) \end{cases}$$

Here, corresponding status, cost and environment requirements of task are also decomposed with goal.

4.2 Organizing matrix

Based on environment-aware measurement and service capacity evaluation, we can establish three kinds of matrix for service organizing: Capacity matrix, environment matrix and social matrix. Suppose a task vector $\boldsymbol{K}=[T_1, T_2, \cdots, T_n]$ and node vector $\boldsymbol{Z}=[d_1, d_2, \cdots, d_m]^{\mathrm{T}}$, and the three kinds of $\boldsymbol{K}\times\boldsymbol{Z}$ matrix can be defined as follows.

Definition 9 Capacity matrix can be defined as

$$\boldsymbol{\Gamma}=\begin{bmatrix}\text{evaluate}_{11} & \cdots & \text{evaluate}_{1m}\\ \vdots & & \vdots\\ \text{evaluate}_{n1} & \cdots & \text{evaluate}_{nm}\end{bmatrix} \tag{16}$$

where value of evaluate_{ij} comes from algorithm 2.

Likewise, we define the schema of environment measurement as environment matrix.

Definition 10 Environment matrix can be defined as

$$\boldsymbol{\Psi}=\begin{bmatrix}\text{environment}_{11} & \cdots & \text{environment}_{1m}\\ \vdots & & \vdots\\ \text{environment}_{n1} & \cdots & \text{environment}_{nm}\end{bmatrix} \tag{17}$$

where value of environment_{ij} can be calculated as follows

$$\text{environment}_{ij}=\begin{cases} p\_\text{value}(d_j) & \text{node}_j.I \wedge \text{node}_j.Ps.\Im\_\text{real}\rightarrow T_i.\text{Envir}\\ \dfrac{1}{2}\times p\_\text{value}(d_j) & \text{node}_j.I \wedge \text{node}_j.Ps.\Im\_\text{mobile}\rightarrow T_i.\text{Envir}\\ 0 & \text{else}\end{cases} \tag{18}$$

Definition 11 Social constraint matrix can be defined as

$$\boldsymbol{\Omega}=\begin{bmatrix}\text{social}_{11} & \cdots & \text{social}_{1m}\\ \vdots & & \vdots\\ \text{social}_{n1} & \cdots & \text{social}_{nm}\end{bmatrix} \tag{19}$$

where value of social_{ij} can be calculated as

$$\text{social}_{ij}=\begin{cases}1 & \text{node}_j.S\rightarrow T_i.\text{Cons}\\ 0 & \text{else}\end{cases} \tag{20}$$

4.3 Self-organizing mechanism for CPS service

As aforementioned, self-organizing aims to find out competent nodes to form a temporary organization in CPS and solve special tasks. We consider that CPS service organizing mechanism is established based on approaches of capacity optimal estimation and task goal solving. Our approach in this paper combines features of CPS from both cyber service and physical environment aspects.

Notice that our proposed capacity matrix, environment matrix and social constraint matrix show the capacity, environment and social constraint matching schema between task and nodes. Therefore, we propose the service self-organizing mechanism of CPS based on the above matrices, which consists of 12 steps as in algorithm 4.

Algorithm 4 Self-organizing mechanism of CPS

Step 1 Task$\leftarrow T_i$; Candidate$\leftarrow\varnothing$; Selected $\leftarrow\varnothing$.

Step 2 CPS delivers the task set Task to nodes which locate in its community.

Step 3 Each node d_j, which receives the task semantic, transmits the task semantic to its neighbors, d_k, which have relationship semantic $d_j.S.r(d_k)$ with d_i and satisfy the constrains of $d_j.S\rightarrow T_i.\text{Cons}$. Then, d_j sends information to CPS sponsor for replying whether it would enter the service organizing and be a candidate node.

Step 4 For task $\boldsymbol{T}_i\in$ Task and ultimate candidate nodes of CPS $\boldsymbol{Z}=[d_1, d_2, \cdots, d_m]^{\mathrm{T}}$, CPS calculates the matrices of $\boldsymbol{\Gamma}$, $\boldsymbol{\Psi}$ and $\boldsymbol{\Omega}$.

Step 5 For $T_i\in$ Task, if there is a node d_l ($d_l\in\boldsymbol{Z}$) with $(\text{capacity}_{il}>0)\wedge(\text{environment}_{il}>0)\wedge(\text{social}_{il}=1)$, it denotes that task T_i can be solved by CPS. CPS selects the node d_l with value of $\max[\text{capacity}_{il}+\text{environment}_{il}]\wedge(\text{social}_{il}=1)$, and Candidate$\leftarrow T_i|d_l$, Task$\leftarrow$Task$-T_i$.

Step 6 For each $T_i \mid d_l\in$ Candidate, CPS calculates the value of competence parameter as

$$\text{competence}(d_l)=\frac{d_l.Cs.\text{price}}{T_i.\text{cost}}+\frac{d_l.Cs.\Im.P}{T_i.\text{Envir}.P} \tag{21}$$

Step 7 For $T_i\in$ Candidate, d_l with the $\min[\text{competence}(d_l)]$ is selected, and Select$\leftarrow$ $T_i|d_l$, Candidate$\leftarrow$Candidate$-T_i|d_l$.

Step 8 Repeat steps 6, 7 until Candidate$=$ $\varnothing$.

Step 9 If Task$\neq\varnothing$, CPS decomposes tasks through algorithm 3 for each $T_i\in$ Task into a new task set dec_T_i.

Step 10 Task$\leftarrow\varnothing$; Task$\leftarrow$dec_T_i.

Step 11 Repeat steps 2 to 10 until Task$=\varnothing$.

Step 12 Return selected.

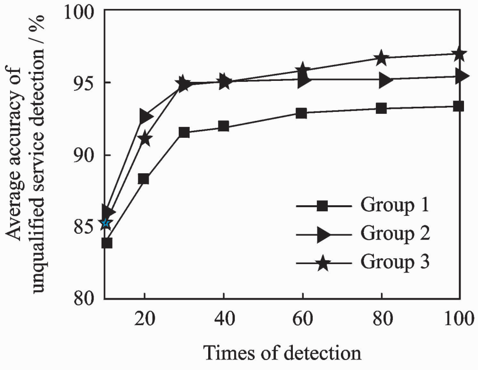
For complicated task, our self-organizing mechanism aims to algorithm 4 to find out a node composition of CPS which meets not only both physical and cyber requirement but with the minimum costs and the maximum service capacities.

5 Examination

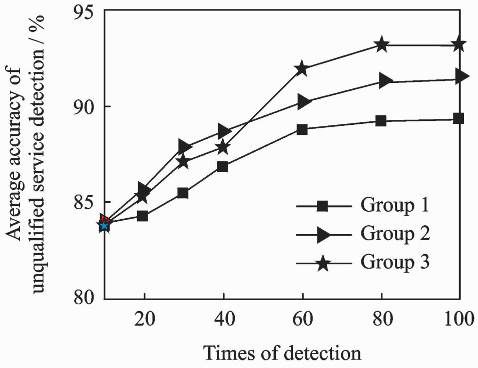
In this section, we employ a set of examinations for performance comparison to testify the feasibility of our proposed method. We utilize our peer-to-peer network simulator which is written in Java language. In our examination, there are 300 nodes in 5 working communities and the average out-degree of node is 5. Each our-degree of node means a social relationship between two nodes. As initial setting, physical and cyber capacity values of all nodes follow a normal distribution with mean 0.6 and variance 0.1. The capacity semantics of nodes, including physical and cyber aspects, are generated at random in advanced. Parameter, β, is set as 0.8.

5.1 Unqualified service detection

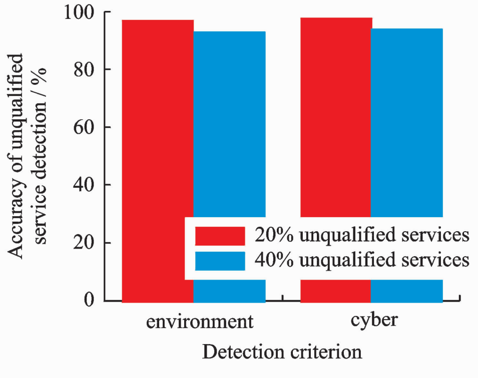
In our proposed method, environment measurement and service capacity evaluation can detect unqualified services according to past data. Here, we conduct a set of examinations to testify the effects of unqualified service detection. We introduce with 20% and 40% unqualified services in our examination. In this examination, services are regarded as unqualified ones while their scores given by different methods are lower than 0.3. We adopt three groups of tests as follows: Group 1 utilizes the average judgments of service for scoring service, Group 2 utilizes the Bayesian rating method and Group 3 adopts our proposed method as scores of p_value and c_value (any one score lower than 0.3 is regarded as unqualified service). We record the average accuracies of unqualified service detections in different three groups. We can see the effects of unqualified service detection in following Figs. 1(a,b) and the average accuracies of our method are about 97% and 93% in two cases of examination. Our method is obviously better than other two ones. However, we notice that the effect of our method is worse than other method at the beginning of examinations. In our consideration, the reason is that our method is proposed based on past data of service and the accuracy will be much better with the detection times growing up. In addition, Fig. 1(c) shows the accuracies of unqualified service detection in two different detection criteria: Environment and cyber. Our proposed method can find



(a) 20% unqualified services



(b) 40% unqualified services



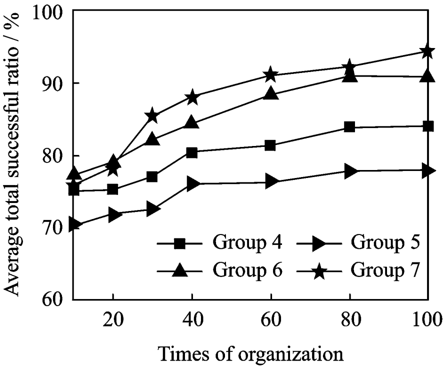
(c) Unqualified service detection

Fig. 1 Effects on unqualified service detection out the unqualified service from both physical environment and cyber capacities aspects.

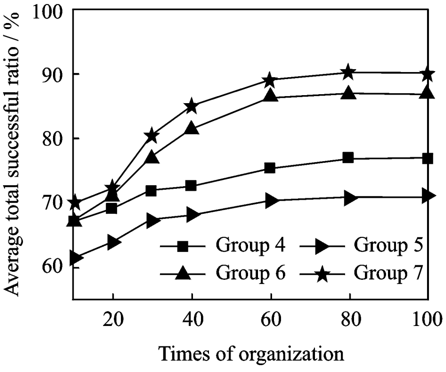
5.2 Successful ratio of service organization

In this examination, we set four groups of service organization method for effect comparison. In groups 4 and 5, services are organized based on services which satisfies the task with highest values of physical environment and cyber capacities respectively. In group 6, services which have the highest selecting probabilities for the corresponding tasks in the past are selected for the new arriving complicated task. In group 7, we utilize our method to organize services for complicated task. We repeat the organization 100 times. Figs. 2(a,b) show the average total successful ratios with 10% and 30% unqualified services respectively. We can see that our method is obviously better than other methods. The average total successful ratios in group 7 are about 94.5% and 90.1% respectively.

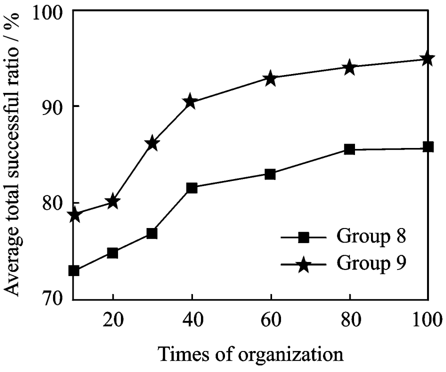
In addition, we test the effect of task decomposition. We repeat service organization 100 times in two groups and the results are shown in Fig. 2(c). Group 8 adopts our proposed mechanism without task decomposition method while Group 9 includs the method. We can see that the successful ratio is raised about 9.5% in group 9. The task semantic decomposition algorithm can decompose the complicated task into sub-task semantics for finding out services to solve them. From this point, task in group 9 has more probabilities to find corresponding services. Such result manifests that our proposed task semantic decomposition method is efficient for service organization.



(a) 10% unqualified services



(b) 30% unqualified services



(c) Effect of task decomposition ratio

Fig. 2 Effects on successful ratio of service organization

6 Conclusions

Different from service composition in internet environment, CPS faces the physical world limitations for its self-organizing service providing. To solve the problem, we address a physical environment sensitive and cyber capacity estimable mechanism for CPS service organizing. Our mechanism is based on formal semantic descriptions and theory of Horn clause, which are introduced for providing system readable information and reasoning rules. By presenting the methods of environment aware measurement and service capacity evaluation, we define task semantic decomposition for complicated task solving feasible and organizing matrix for identifying the matching degrees between CPS nodes and tasks. Finally, we propose the self-organizing mechanism for CPS to find out the most competent service composition. We plan to examine our methodology in simulation to verify feasibility and efficiency in future. Furthermore, we are utilizing our methodology and framework into smart connected cars application.

References:

[1] Tan Y, Goddard S, Perez L C. A prototype architecture for cyber-physical systems[J]. ACM SIGBED Review, 2008,5(1):1-2.

[2] Westerman D, Spence P R, Brandon Van Der Heide. A social network as information: The effect of system generated reports of connectedness on credibility on Twitter [J]. Computers in Human Behavior, 2012,28:199-206.

[3] Song Zhen, Chen Yangquan. Feasibility analysis on optimal sensor selection in cyber-physical systems [C]//2009 American Control Conference Hyatt Regency Riverfront. St. Louis, MO, USA: IEEE, 2009:5368-5373.

[4] Lee E A. Cyber physical systems: Design challenges [C]//11th IEEE Symposium on Object Oriented Real-Time Distributed Computing. Orlando, Florida, USA: IEEE, 2008:363-369.

[5] Dabholkar A, Gokhale A. An approach to middleware specialization for cyber physical systems[C]// 29th IEEE International Conference on Distributed Computing Systems Workshops. Montreal, Québec, Canada: IEEE, 2009:73-79.

[6] Antsaklis P. On control and cyber-physical systems: Challenges and opportunities for discrete event and hybrid systems[C]//9th International Workshop on Discrete Event Systems. Goteborg, Sweden: IEEE, 2008:1.

[7] Cardenas A A, Amin S, Sastry S. Secure control: Towards survivable cyber-physical systems[C]//Proceedings of the 28th International Conference on Distributed Computing Systems Workshops. Beijing, China: IEEE Society, 2008:495-500.

[8] Xia Feng, Ma Longhua, Dong Jinxiangt, et al. Network QoS management in cyber-physical systems [C]//Proceedings of International Conference on Embedded Software and Systems Symposia. Chengdu, China: IEEE Society, 2008:302-307.

[9] Kang K D, Son S H. Real-time data services for cyber physical systems[C]//Proceedings of 28th International Conference on Distributed Computing Systems Workshops. Beijing, China: IEEE, 2008: 483-488.

[10] Buczak A L, Wang H H, Darabi H, et al. Genetic algorithm convergence study for sensor network optimization[J]. Information Sciences, 2001, 133(3): 267-282.

[11] Bian F, Kempe D, Govindan R. Utility-based sensor selection[C]//Proceedings of International Conference on Information Processing in Sensor Networks (IPSN). Nashville, TN: IEEE, 2006:11-18.

[12] Li Dongsheng, Lv Qin, Xie Xing, et al. Interest-based real-time content recommendation in online social communities [J]. Knowledge-Based Systems, 2012,28:1-12.

[13] Westerman D, Spence P R, Brandon Van Der Heide. A trust calculating algorithm based on social networking service user′s context[J]. Chinese Journal of Computers, 2011,34(12):2403-2413.

[14] Alfred Horn. On sentences which are true of direct unions of algebras[J]. Journal of Symbolic Logic, 1951,16:14-21.

[15] Lautenbach K. Logical reasoning and petri nets[C]// Proceedings of the ICATPN 2003. Berlin, Heidelberg: Springer-Verlag, 2003, LNCS 2679:276-295.

(Executive editor: Xu Chengting)