Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2012, Article ID 242654, 11 pages
doi:10.1155/2012/242654

Research Article

A Novel Reliability Assurance Method for Cyberphysical System Components Substitution

Peng Wang,[1, 2] Yang Xiang,[1] and Shaohua Zhang[2]

[1] *College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China*
[2] *Shanghai Development Center of Computer Software Technology, Shanghai 201112, China*

Correspondence should be addressed to Peng Wang, haris2003@163.com

Received 3 August 2012; Revised 4 October 2012; Accepted 8 October 2012

Academic Editor: Liguo Zhang

Reliability of cyberphysical system (CPS) components substitution is an important issue for CPS troubleshooting and system upgrading. In this paper, decision problem of components substitution is regarded as decision problem of services substitution through a service-oriented architecture of CPS. Further, a reliability assurance method for CPS service substitution is proposed, which comprises two parts. The first one is a qualitative judgment method for CPS service substitution according to the relationship between service compatibility and substitution based on time-space π-calculus with time and space operators. The other one is consisted of substitution processes from above judgment results based on service management theory. Finally, a case study is performed to show how to apply this method to ensure CPS components reliable substitution. The experimental result shows that this method is reasonable and feasible.

1. Introduction

Cyberphysical system (CPS) is a new concept in the information field in recent years. CPS is defined [1] as integration of computation with physical processes and consists of computation units, control units, communications network, sensors, and actuators. In a CPS, downsized and embedded devices execute physical processes by monitoring and controlling entities in the physical world. Computers, networks, devices, and their environments in which they are embedded have interacting physical properties, consume resources and contribute to the overall system behavior. Nowadays, CPS can be found in areas as diverse as transportation, defense, energy, and industrial automation, health, biomedical and critical infrastructure, agriculture, and so forth. Many countries have begun to pay high attention to CPS [2]. The PCAST of USA in 2007 report found that cyberphysical systems "are now a national priority for Federal R&D." Hundreds of millions of dollars are invested into R&D efforts from then on. The European Union's Artemis (2008–2017) is clearly aimed at the same fundamental problems in the embedded systems aspect of CPS research, with €2.7 billion. Others such as Japan and Korea have set up CPS research projects. Chinese government also attaches great importance and the 863 program "CyberPhysical Oriented System Platforms" has started, officially approved in 2011.

Since CPS has impact on physical processes and an unreliable operation may lead to disastrous consequences, CPS components' reliable substitution is an important issue for troubleshooting and system upgrading. The first problem is how to design the system architecture of CPS. However, the research on architecture is still at knowledge preliminary and exploratory stage both at home and abroad. Tan Ying proposed prototype architecture of CPS [3], but it lacks a comprehensive and deep description of the layers. Phan and Lee presented an approach towards a compositional multimodal framework of CPS [4], but composition analysis has been limited to uniprocessor processing elements and EDF/FP scheduling policies. Koubaa and Andersson provided a realistic vision to the concept of the Cyberphysical Internet [5], but it does not solve the problem of real time for CPS. In this paper, a service-oriented architecture [6] of CPS is put forward, in which software and hardware of CPS are designed and developed in the form of interoperable services. Based on this architecture, components substitution is equated to CPS service substitution, which is easy to realize formal analysis.

There are many formal modeling research on CPS, including Petri Net (PN) [7], Finite State Machines (FSM) [8], Process Algebra [9]. PN and FSM are intuitive to be analyzed by using charts. However, the major problem is state-space-explosion. Π-calculus [10] is chosen in the paper which is a process calculus as a formal analysis tool, for its ability to describe concurrent computations whose network configuration may change during the computation like CPS. Since CPS is bound by time and space (position, energy) constraints, time-space π-calculus is proposed through introducing time operator and space operator into π-calculus. And a formal method for modeling CPS service is presented based on it. Starting with the relationship between service compatibility and substitution [11], a qualitative judgment method for CPS service substitution is put forward. Then, from process management perspective, a series of substitution processes are advanced from the judgment results above, referring to the best practice and international standard of service management (such as ITIL [12], ISO20000 [13]). There are some worthwhile research on service substitution management [14, 15], and in this paper, the substitution processes combine service substitution requester, service substitution implementer, substituted CPS service and CPS, so that service substitution can be implemented according to standardized and normalized process.

The remainder of this paper is organized as follows. In Section 2, the conceptions of CPS service and CPS service compatibility are described in detail. In Section 3, a reliability assurance method is proposed, which consists of two parts, that is, a qualitative judgment theorem via Time-Space π-calculus for CPS service substitution and a series of substitution processes from the judgment results. In Section 4, a case study, Electronic Fence, is performed to show how to apply this method to ensure CPS components' reliable substitution. Finally, the conclusion is given.

2. Basic Conceptions

2.1. CPS Service. In this section, a service-oriented architecture is proposed that distributed and open-ended CPS is regarded as a combination of encapsulated CPS service, following some business logic and business processes. The service-oriented architecture is shown in Figure 1. In this framework, CPS is divided into four layers: application layer, business process layer, service abstraction layer, and service implementation layer. Each layer is described as follows.

2.1.1. Service Implementation Layer. Service implementation layer is the foundation of this architecture, and it is also the implementation of CPS service interface. Details about how to implement it are hidden for service users, and different service providers can use different technology to implement the same service interface. Each CPS service implementation contains sense-actuate unit, communications unit, and computation-control unit. Sense unit monitors physical world and transfers monitoring information to computation unit through communications unit. Then computation unit determines strategies and sends them to control unit. Control unit gives instructions to actuate unit through communications unit, to control physical processes. Each unit is described as follows.

(1) Sense-actuate unit contains sensors, actuators, and terminal computation module. Sensors monitor physical entities and physical environment. Actuators control physical processes. Terminal computation module contains basic executive rules of actuator and has small storage capacity of real-time data.

(2) Communications unit provides ubiquitous communication mechanism by fusing 2G, 3G, 4G, and so forth. This unit also involves real-time interaction, integration of heterogeneous networks, security of communications, and communication quality.

(3) Computation-control unit contains computation unit and control unit. Computation unit mixes discrete domain and continuous domain together. Control unit implements strict management to time and space. Strategies determined by this unit can be supported by cloud computing center and knowledge base.

2.1.2. Service Abstraction Layer. Service abstraction layer defines service functions accessed outside and how to access them. However, this layer does not contain the details about how to implement them. This layer also involves service description, service registry, service discovery, and quality of service. Specially, it describes interfaces' characteristics, operation's usability, parameters, data type, and access protocol. Through this layer, services or modules outside know what CPS service can do, how to find it, how to exchange message, how to invoke it, and what returned results may be. Specially, there must be physical properties in CPS service description, for example, timestamp, position information and energy information of physical entities, for service implementation layer contains physical unit (computation unit and control unit) and monitored information without temporal and spatial information is meaningless. There are two types of services in this layer, that is, business service and infrastructure service. Each type is described as follows.

(1) Business service is part of business process and fine-grained subprocess of business requirement. It can fulfill a specific business task automatically and can be reused among different business processes. It is of two kinds: business function service and common service. The former is related to some business area for example, real-time positioning, driver monitor, and remote alarm in an intelligent transportation CPS. The latter one can be used in different business areas for example, common algorithm, data transformation, and so forth.

(2) Infrastructure service is the foundation of standardized integration of CPS service. It involves time synchronization, space constraints, general technology,

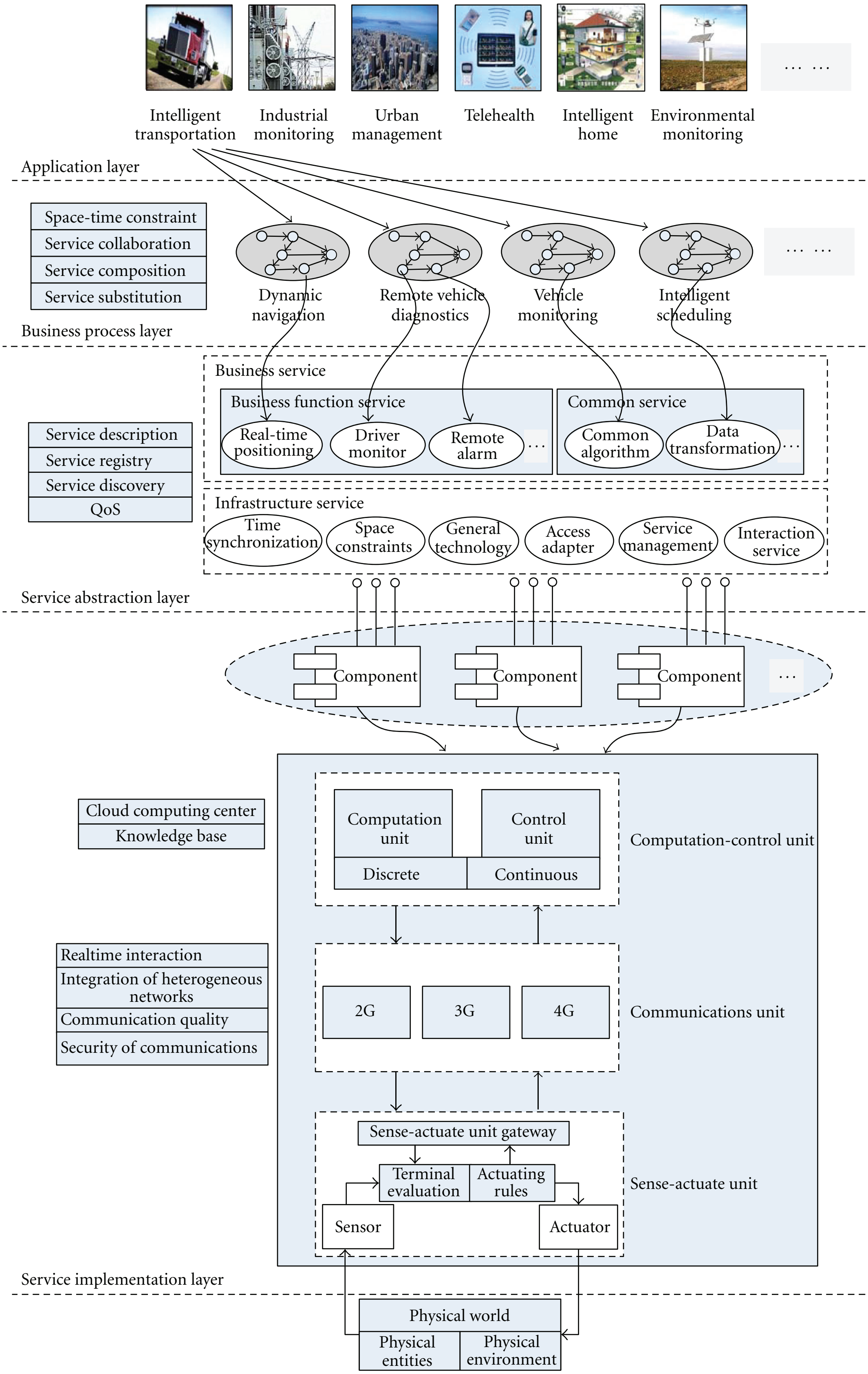


Figure 1: SOA framework of CPS.

access adapter, service management, and interaction service. Time synchronization and space constraints are guaranteed to meet the temporal and spatial condition when physical units and cyber units are mixed together in multiple scales. General technology provides technology infrastructure for developing, delivering, maintaining CPS service, as well as the abilities of security, performance, availability, and so forth. Access adapter changes available resources of legacy systems into individual business service. Service management is to monitor CPS service's state and provide support for abnormal condition for example, SLA, capacity planning, cause analysis, and so forth. Interaction service is used for arranging interfaces of CPS service into intelligent device, not only for human-computer interaction.

2.1.3. Business Process Layer. Business process layer involves a number of business processes, where each business process is composed of CPS services following regular rules. It is necessary to set up a properly complicated and reliable layer like this, since a lot of fine-grained CPS services will lead to great cost and be ineffective. This layer also involves service collaboration, service composition, service substitution, and space-time constraints.

2.1.4. Application Layer. Application layer involves many industry applications of CPS, in which each system is composed of business processes. These business processes are cooperated with each other in order to fulfill higher level business goals. Compared with business process layer, this layer tend to be more focused on integrating all kinds of application requirements from combining professional knowledge with business model in different industries.

Interface of CPS service, containing interface characteristics, operation usability, parameters, data type and access protocol, is implemented with component technology. Service users can know what CPS service can do, how to find it, how to exchange message, how to invoke it, and what may returned results be through interface. However, details about how to implement it are hidden; therefore, service providers can implement a same service interface by different technologies. Since CPS service provides interface to receive and send messages and transit from initial state to final state by triggering of send-receive actions. Meanwhile, it takes time and consumes energy to complete these actions. Let us give the definition of CPS service view.

Definition 1 (CPS service view). A CPS service view is defined as nine tuples: $\mathrm{CPSV} = (S, s_0, F, \mathrm{Act}, T, M, f_m, f_t, f_e)$, where $S = \{s_0, s_1, \ldots\}$: set of finite states.

s_0: Initial state of CPS service.

F: Final states set of CPS service, $F \subseteq S$.

$\mathrm{Act} = A \cup \overline{A} \cup \{\tau\}$: Set of actions.

$A = \{a \mid a \in \Sigma\}$ is set of receiving actions (Σ is alphabet).

$\overline{A} = \{\overline{a} \mid a \in \Sigma\}$ is set of sending actions.

τ is internal action.

$T \subseteq S \times \mathrm{Act} \times S$: State transitions relation.

M: Set of CPS service messages.

$f_m : \mathrm{Act} \to M$ action-message function. For all $a \in \mathrm{Act}$, $f_m(a)$ are receiving (resp. sending) messages of a.

f_t: $\mathrm{Act} \to R^+$ action-time function. For all $a \in \mathrm{Act}$, $f_t(a)$ is the time spending on completing a.

f_e: $\mathrm{Act} \to R^+$ action-energy function. For all $a \in \mathrm{Act}$, $f_e(a)$ is the energy consuming to complete a.

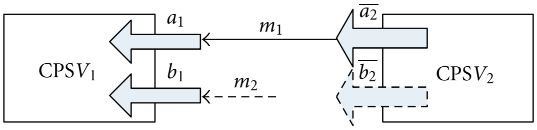


Figure 2: Interaction between CPSV$_1$ and CPSV$_2$.

2.2. Compatibility of CPS Service. Performing complex business tasks typically needs to make a number of CPS services work together. It is therefore necessary to ensure that these services are able to interact properly, which is the notion of compatibility. Compatibility is aimed at interactive processes of CPS services. From the aspect of CPS service view, an interactive process represents a series of calls between two CPS services. When one CPS service sends (resp., receives) message, this means that the other CPS service simultaneously evolves by receiving it (resp., sending it). So in a sense the behavior of CPS service 2 should be the same as CPS service 1, but with receptions instead of emissions, and vice versa. The dual service $\overline{S}$ of CPS service S is defined that when the emissions are changed to receptions and vice versa, and the notation $\overline{a}$ represents opposite action of action a. Let us define interaction element, normal interaction element and abnormal interaction element as follows.

Definition 2 (interaction element/normal interaction element/abnormal interaction element). There are two CPS service views $\mathrm{CPSV}_1 = (S_1, s_{0_1}, F_1, \mathrm{Act}_1, T_1, M_1, f_{m_1}, f_{t_1}, f_{e_1})$ and $\mathrm{CPSV}_2 = (S_2, s_{0_2}, F_2, \mathrm{Act}_2, T_2, M_2, f_{m_2}, f_{t_2}, f_{e_2})$. An interaction element is defined as three tuples $ie = (a_1, \overline{a}_2, m)$.

(1) When $a_1 \in A_1$, $\overline{a}_2 \in \overline{A}_2$, $a_1 = \overline{\overline{a}}_2$, $m = f_{m_1}(a_1) = f_{m_2}(\overline{a}_2)$, ie is called normal interaction element.

(2) When $a_1 \in A_1$, $\overline{a}_2 \in \varnothing$, $m = f_{m_1}(a_1)$, ie is called abnormal interaction element.

Interaction element represents a step of interaction between two CPS services. Normal interaction element represents a successful interaction, in which the two interactive actions are dual with a same receiving (resp., sending) message. Abnormal interaction element represents an unsuccessful interaction, in which one CPS service has receiving action but the other does not have sending one. As shown in Figure 2, there are two interaction elements, that is, $ie_1 = (a_1, \overline{a}_2, m_1)$; $ie_2 = (b_1, \overline{b}_2, m_2)$, ie_1 is a normal interaction element, and ie_2 is an abnormal interaction element.

Compatibility between two CPS services arises at different levels, that is, static compatibility and dynamic compatibility. Static compatibility is the semantic and syntactic compatibility. Dynamic compatibility is that exchanges of messages are ordered in matched sequences without deadlock and livelock, and there are no sending messages that cannot be received by one of the two CPS services. Assuming that CPS service A and CPS service B are static compatible, and sending messages set of A is a subset of receiving messages set of B (i.e., A partially or fully uses the receiving message interfaces of B), if A and B are able to interact properly, they are called being compatible. Let us give the formal definition of compatible.

Definition 3 (compatibility degree). IE_n represents set of a CPSV's normal interaction elements, and $N(IE_n)$ represents the number of elements in IE_n. IE_a represents set of the CPSV's abnormal interaction elements, and $N(IE_a)$ represents the number of elements in IE_a. Compatibility degree is defined as $\omega = N(IE_n)/(N(IE_n) + N(IE_a))$.

Definition 4 (fully compatible/partially compatible/incompatible). Let M denote set of other CPSVs interacting with this CPSV, ω denote compatibility degree. If $\omega = 1$, CPSV and M are fully compatible. If $0 < \omega < 1$, they are partially compatible. If $\omega = 0$, they are incompatible. Fully compatible and partially compatible are referred to as compatible.

In Figure 2, CPSV$_1$ and CPSV$_2$ are partially compatible, and $\omega = 1/2$.

3. Reliability Assurance Method for CPS Service Substitution

In this section, time-space π-calculus is proposed to model CPS service. Then, a reliability assurance method for CPS service substitution is put forward, which consists of two parts, that is, a qualitative judgment theorem for CPS service substitution and a series of substitution processes from the judgment results.

3.1. Time-Space π-Calculus. CPS service has a good corresponding relationship with process of π-calculus. Specifically, communication channels of process represent actions of CPS service, sending-receiving variables of process represent sending-receiving messages of actions, and process, summation, composition, replication in π-calculus represent sequence structure, case structure, parallel structure, and iterative structure of CPS service composition. However, π-calculus lacks syntax about time and space characteristics. So we put forward the notion of time-space π-calculus, through introducing time and space (position, energy) operators into π-calculus.

Since relative accuracy of the time is enough to meet quality of CPS service requirement, discrete time domain is adopted to describe time characteristic of CPS in this paper. Properties of discrete time domain are defined as follows.

Definition 5 (properties of discrete time domain). Discrete time domain T has following properties.

(1) For all $t \in T$, $t \neq 0 \Rightarrow t > 0$;

(2) for all $t \in T$, $t \neq \infty \Rightarrow \infty > t$;

(3) for all $t, t' \in T$, $t > t' \Leftrightarrow \exists \Delta t > 0, t' + \Delta t = t$;

(4) for all $t, t' \in T$, $(t > 0) \wedge (t' \neq \infty) \Rightarrow t' + t > t'$;

(5) for all $t \in T$, $t + 0 = t, t + \infty = \infty$;

(6) for all $t_1, t_2 \in T$, $t_1 > t_2$, $\{t \mid t_1 \leq t \leq t_2\}$ is expressed as $[t_1, t_2]$, which is called time interval;

(7) for all $t_2, t_3 \in T$, for all $[t_1, t_4]$, $\exists t', t' \in T$, $t_2 \leq t' \leq t_4$, then $t' \in [t_1, t_2]$.

Definition 6 (time operator $\equiv$ $\mathrm{Int}(t_r, \Delta t)$). t_r is benchmark time, $\Delta t \geq 0$. $\mathrm{Int}(t_r, \Delta t)P$ represents that process P can start only when it meets $t \in [t_r, t_r + \Delta t]$.

Physical components of CPS are abstracted to spatial objects based on OGC [16] (Open Geospatial Consortium) and topological relation theory of spatial database [17]. The topological relations between two spatial objects, which are regarded as point sets, are expressed by a quaternion formed by boundary and interior of point set. Here, A and B represent two spatial objects. Let $\partial A, A^0, \partial B, B^0$ denote boundary and interior of A and B. The quaternion is $\mathbf{R}(A, B) = \left(\begin{smallmatrix} \partial A \cap \partial B & \partial A \cap B^0 \\ A^0 \cap \partial B & A^0 \cap B^0 \end{smallmatrix}\right)$. Topological relations include eight kinds, that is, disjoint, meet, equal, overlap, inside, contain, covered by, and cover, which are represented by $S_{\mathrm{pos}} = \{s_d, s_m, s_e, s_o, s_{in}, s_{ct}, s_{cb}, s_c\}$. Assuming that process P contains n physical components and c_i represents the relation between the ith physical component and benchmark region, $\exists S_i \subseteq S_{\mathrm{pos}}$, this physical component can function properly only when it meets $c_i \in S_i$. Let $S = \{S_1, S_2, \ldots S_n\}$. Position operator is defined as follows.

Definition 7 (position operator $\equiv$ $\mathrm{Pos}[S]$). $\mathrm{Pos}[S]P$ represents that process P can start only when truth-value of $\prod_{i=1}^{n} c_i \in S_i$ is true.

All the observable energy, which supports physical components of CPS functioning well, is called energy information of CPS. It includes many kinds, for example, electric energy, heat energy, and so forth, and can be consumed and replenished. Assuming that process P contains n physical components, E_i represents energy value of the ith physical component, $E_{i\max}$ represents the maximum energy value of P, $\exists m_i \in [0, 100]$, this physical component can function properly only when it meets $E_i \geq (E_{i\max} \cdot m_i\%)$. Let $M = \{m_1, m_2, \ldots, m_n\}$. Energy operator is defined as follows.

Definition 8 (energy operator $\equiv$ $\mathrm{Ene}[M]$). $\mathrm{Ene}[M]P$ represents that process P can start only when truth-value of $\prod_{i=1}^{n} E_i \geq (E_{i\max} \cdot m_i\%)$ is true.

Definition 9 (syntax of time-space π-calculus).

$$\begin{aligned} P ::= {} & 0 | \overline{a}\langle x\rangle \cdot P | a(x) \cdot P | \tau \cdot P | P \\ & + Q | P \mid Q | (x) P \mid [x = y] P \mid !P | \mathrm{Int}(t_r, \Delta t) P | \\ & \times \mathrm{Pos}[S] PI. \end{aligned} \tag{1}$$

0 is nil process. $\overline{a}\langle x\rangle \cdot P$, $a(x) \cdot P$, $\tau \cdot P$ are output prefix, input prefix, and silent prefix process. $P + Q$ is sum process. $P \mid Q$ is concurrency composition process. $(x)P$ is restriction process. $[x = y]P$ is match process. $!P$ is replication process. Detailed meanings of the nine expressions above can be seen in [10]. Meanings of the last three can be seen in Definitions 6, 7, and 8.

Definition 10 (operational semantics of time-space π-calculus).

(1) Time operator ($\alpha \in \{\tau, a(x), \overline{a}\langle x\rangle\}$, other "$\alpha$" below are identical with this one). Consider

$$\frac{P \xrightarrow{\alpha} P'}{\mathrm{Int}(t_r, \Delta t)P \xrightarrow{\alpha} P'}, \quad t \in [t_r, t_r + \Delta t],$$
$$\frac{P \xrightarrow{\alpha} P'}{\mathrm{Int}(t_r, \Delta t)P \xrightarrow{\alpha} 0}, \quad t \notin [t_r, t_r + \Delta t]. \tag{2}$$

(2) Position operator. Consider

$$\frac{P \xrightarrow{\alpha} P'}{\mathrm{Pos}[S]P \xrightarrow{\alpha} P'}, \quad c_i \in S_i; \qquad \frac{P \xrightarrow{\alpha} P'}{\mathrm{Pos}[S]P \xrightarrow{\alpha} 0}, \quad c_i \notin S_i. \tag{3}$$

(3) Energy operator. Consider

$$\frac{P \xrightarrow{\alpha} P'}{\mathrm{Ene}[M]P \xrightarrow{\alpha} P'}, \quad E_i \geq (E_{i\max} \cdot m_i\%),$$
$$\frac{P \xrightarrow{\alpha} P'}{\mathrm{Ene}[M]P \xrightarrow{\alpha} 0}, \quad E_i < (E_{i\max} \cdot m_i\%). \tag{4}$$

Operational semantics of PREFIX, SUM, PAR, COM, MATCH, RES, and OPEN can be seen in [10].

The performance influence of time-space π-calculus is poorer than classical π-calculus due to the additional time and space operators. Fortunately, the deduction procedures can be completed automatically by a software tool of π-calculus—MWB [18].

Definition 11 (weak simulation/weak bisimulation). Let R denote a binary relation in processes domain K. For all $(P, Q) \in R$, $P \in K$, $Q \in K$, if the following conditions are satisfied, Q is said to be weak simular with P.

(1) Whenever $P \rightarrow P'$, then $\exists Q' \in K$ such that $Q \Rightarrow Q'$ and $(P', Q') \in R$.

(2) Whenever $P \xrightarrow{\lambda} P'$, then $\exists Q' \in K$ such that $Q \overset{\lambda}{\Rightarrow} Q'$ and $(P', Q') \in R$.

If symmetric requirements with P and Q interchange, the relationship between P and Q is said to be weak bisimulation, written $P \approx Q$.

Properties of weak bisimulation can be seen in [10]. Weak bisimulation is used to describe the situation that two processes are equivalent looking outside but have different internal structure and actions.

3.2. Qualitative Analysis Method for CPS Service Substitution.

Substitutability is closely related to compatibility. Combining related research results, sufficient conditions of CPS service substitution are proposed. Let S, S' denote two CPS services. If S' is compatible with all CPS services which are compatible with S, sending-receiving messages set of S is subset of S', and S' can meet time and space constraints, then S' can be substituted for S.

Let $\mathrm{CPSV}_{S'}$ denote CPS service view of S' which has n interaction elements, $T_r = \{t_{r1}, t_{r2}, \ldots t_{rn}\}$ and $\Delta T = \{\Delta t_1, \Delta t_2, \ldots \Delta t_n\}$ denote benchmark time set and delay time set of all interactions. And let $P_{S'}$ denote $\mathrm{CPSV}_{S'}$, $P_{\overline{M}}$ denote dual service set, Q_F denote final states set of S', α denote external and internal actions. Then let emissions(S) and receptions(S) denote name sets of sending and receiving messages. Let us give CPS service substitution judgment theorem as follows.

Theorem 12 (CPS service substitution judgment theorem). *Let $M = \{S_1, S_2, S_3, \ldots, S_n\}$ denote set of all CPS services compatible with S. If following conditions are satisfied, S′ can be substituted for S.*

(1) $\mathrm{Int}(T_r, \Delta T)\mathrm{Pos}[S]\mathrm{Ene}[M]P_{S'} \xrightarrow{\alpha} Q_F$;

(2) *S' is static compatible with M;*

(3) $P_{\overline{S'}} \approx P_M$;

(4) emissions(S) $\subseteq$ emissions(S'), receptions(S) $\subseteq$ receptions(S').

Proof. According to condition one and Definitions 6, 7, and 8, S' meets time and space constraints. According to condition two, condition three, Definition 11, and definition of compatibility, S' is compatible with M. And with condition four, the sufficient conditions of CPS service substitution are satisfied—QED.

When Theorem 12 is used in practice, it is easy to decide whether conditions two and four are satisfied. But for condition one and three, we need to build a time-space π-calculus-based ideal model with time and space constraints of CPS. Then, utilizing time and space characteristics of actual CPS service, we can judge whether this process expression is deadlock or livelock, and whether it can reach final state by syntax and operational semantics of time-space π-calculus. □

3.3. Substitution Processes Based on Service Management.

Substitution processes presented in this paper consist of service desk, event management, problem management, change management, configuration management, and knowledge base management. As shown in Figure 3, all substitution requests are accepted by unified service desk, and lifecycle of substitution request is whole monitored. From the analysis results of Theorem 12, according to Definitions 3 and 4, for CPS service incompatible, substitution request is rejected. For $\omega = 1$, event management is adopted to implement service substitution. And for $0 < \omega < 1$, problem management is adopted. Solutions and experience of substitution are shared by knowledge base. Changes of CPS are logged and supervised comprehensively in CMDB (configuration management database). Each process is described as follows.

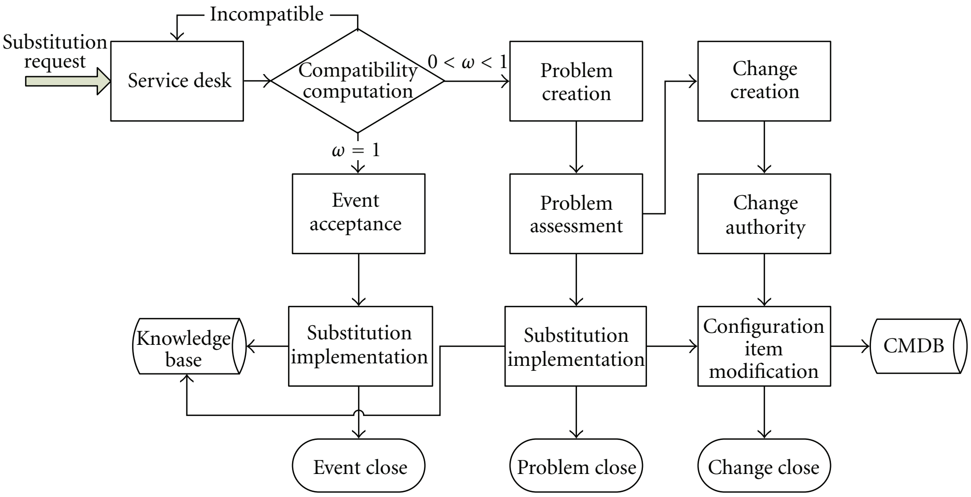


Figure 3: Whole flowchart of CPS service substitution.

3.3.1. Service Desk. Based on this unified access point, all substitution requests are recorded completely and supported preliminarily, and then they are passed to substitution implementer to ensure timeliness of request handing. Service desk can provide accurate process information from start to finish.

3.3.2. Event Management. For requests about CPS services fully compatible, this process provides substitution corresponding service according to SLA.

3.3.3. Problem Management. For requests about CPS services partially compatible, abnormal interaction elements are found out by assessment and analysis in this process. Then, substitution solution is formulated and implemented. Problem management minimizes the effects of abnormal interaction elements to improve service quality and customer satisfaction and also provides support to change management process.

3.3.4. Change Management. This process coordinates with problem management process to implement changes of CMDB. Change management reduces failure rate caused by system changes.

3.3.5. Configuration Management. In this process, description information of CPS service, for example, states, actions, messages, time and space characteristics, and so forth, are centrally managed in CMDB. Configuration management records and controls the changes of CPS.

3.3.6. Knowledge Base Management. This process supports storing, auditing, filtering, updating, and abolishing substitution-related knowledge and accumulates experience about past events and problems solutions.

4. Experiments and Results

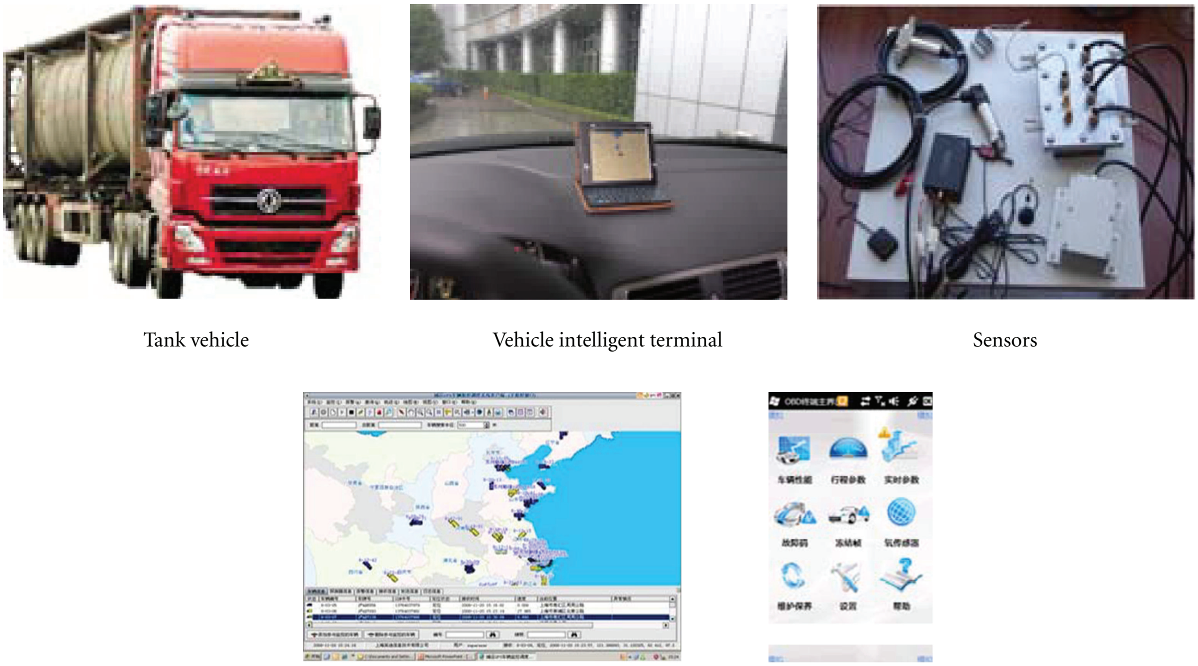
In this section, we take Electronic Fence in hazardous chemicals transport CPS for example (shown in Figure 4), to illustrate how to use the reliability assurance method for CPS components substitution.

4.1. Problem Description. Business case descriptions of electronic fence are as follows. When a tank vehicle loading hazardous chemical products has traffic accident, once sensors in it monitor unusual states, vehicle terminal would send alarms to accident handling center. The center can decide whether it is necessary to set electronic fence by analyzing remote monitoring information, including tank temperature, gas strength, tank liquid level, tank pressure, and so forth. If needed, electronic fence would be set. Then tank vehicles inside the electronic fence are given early warnings periodically. Meanwhile, tank vehicles outside are informed periodically.

Based on the models proposed in Section 2.1, electronic fence is designed to a business process consisting of five CPS services that is, Vehicle Alarm, Remote Diagnosis, Electronic Fence, Early Warning and Accident Told, as shown in Figure 5. Sending and receiving messages are also shown in Figure 5.

Because of user requirements changing, the system upgrades and agent of traffic accident treatment platform is added. Specifically, after remote diagnosing, traffic accident information must be reported to this agent, and when electronic fence setting is completed, electronic fence information must be reported to this agent too. As shown in Figure 6, after upgrading CPS service, that is, agent of traffic accident treatment platform, is added, Remote Diagnosis and Electronic Fence are changed, and other CPS services stay the same.

The problem to be solved is described as that whether S' in Figure 6 can be substituted for S in Figure 5 and how to ensure reliability of this substitution. In order to make this substitution with universality, Electronic Fence is not upgraded to the status in Figure 6 that it cannot send message of ElectronicFenceInf. In such situation, S' is partially compatible with system. Otherwise, if fully upgraded, S' is fully compatible with system, and this situation is idealized without universality.


Tank vehicle

Vehicle intelligent terminal

Sensors

Screenshot of call center system

Screenshot of vehicle terminal system

Figure 4: Real pictures of Electronic Fence in hazardous chemicals transport CPS.

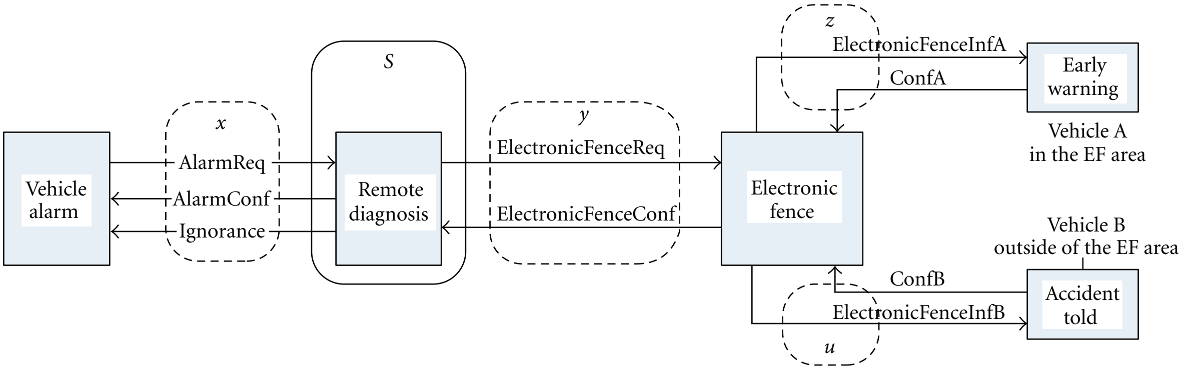


Figure 5: Primal business process of the Electronic Fence.

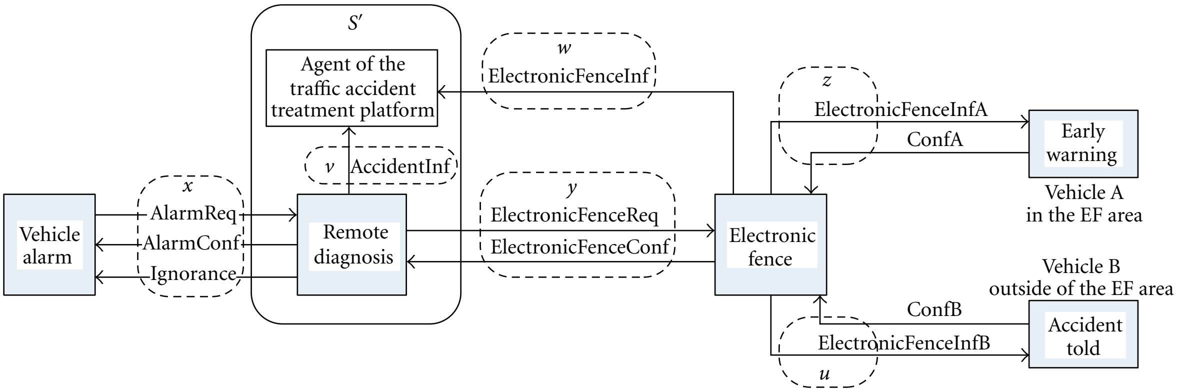


Figure 6: Upgraded business process of the Electronic Fence.

4.2. Problem Modeling. Let $P_{va}, P_{rd}, P_{ef}, P_{ew}, P_{at}$ denote CPS services in Figure 5, with channels names shown in it. In respect of position constraints, let electronic fence be benchmark region. Early Warning is inside of the benchmark that is, $S_{ew} = \{\{s_{in}\}, \{s_{in}\}, \ldots \{s_{in}\}\}$. Accident Told is outside of the benchmark, that is, $S_{at} = \{\{s_d\}, \{s_d\}, \ldots \{s_d\}\}$. The other three CPS services are with no limit, that is, $S_{va} = S_{rd} = S_{ef} = \{S_{\text{pos}}, S_{\text{pos}}, \ldots S_{\text{pos}}\}$. In respect of energy constraints, let $M_{va}, M_{rd}, M_{ef}, M_{ew}, M_{at}$ denote M of Definition 8. Time constraints are described in detail below. Each process expression is modelled as follows.

(1) Vehicle Alarm. Within $t_0 + 1$ seconds (unit below the same) of traffic accident, alarm message must be sent. After that, answer must received in $t_1 + 60$

$$\begin{aligned} P_{va} = {} & \mathrm{Int}(t_0, 1)\mathrm{Pos}[S_{va}]\mathrm{Ene}[M_{va}]\overline{x}\langle \mathrm{AlarmReq}\rangle \\ & \cdot \Big(\mathrm{Int}(t_1, 60)\mathrm{Pos}[S_{va}]\mathrm{Ene}[M_{va}]x(\mathrm{AlarmConf}) \\ & + \mathrm{Int}(t_1, 60)\mathrm{Pos}[S_{va}]\mathrm{Ene}[M_{va}]\Big[S_{\text{pos}}\Big]x(\mathrm{Ignorance})\Big). \end{aligned} \tag{5}$$

(2) Remote Diagnosis. Within $t_2 + 30$ of receiving alarm message, this CPS service must make diagnosis and immediately respond. If confirmed as corresponding accident level, electronic fence application must be submitted within $t_3 + 1$. After that, confirmation must be got in $t_4 + 60$

$$\begin{aligned} P_{rd} = {} & \mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]x(\mathrm{AlarmReq}) \\ & \cdot (\mathrm{Int}(t_2, 30)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]\overline{x}\langle \mathrm{AlarmConf}\rangle \\ & + \mathrm{Int}(t_2, 30)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]\overline{x}\langle \mathrm{Ignorance}\rangle) \\ & \cdot [\mathrm{msg} = \mathrm{AlarmConf}] \\ & \times (\mathrm{Int}(t_3, 1)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]\overline{y} \\ & \times \langle \mathrm{ElectronicFenceReq}\rangle \\ & \cdot \mathrm{Int}(t_4, 60)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]y \\ & \times (\mathrm{ElectronicFenceConf})). \end{aligned} \tag{6}$$

(3) Electronic Fence. Within $t_5 + 30$ of receiving electronic fence application, setup must be finished. Then, electronic fence information must be sent to tank vehicles in different regions in $[t_6, t_6 + 30]$ periodically ($t_6 = t_6 + 30$, increasing constantly). We only take A inside of electronic fence and B outside for example. Confirmation from A must be got in $t_{Ae} + 30$ (t_{Ae} is electronic fence information sending time each time), and so is B

$$\begin{aligned} P_{ef} = {} & \mathrm{Pos}\Big[S_{ef}\Big]\mathrm{Ene}\Big[M_{ef}\Big]y(\mathrm{ElectronicFenceReq}) \\ & \cdot \mathrm{Int}(t_5, 30)\mathrm{Pos}\Big[S_{ef}\Big]\mathrm{Ene}\Big[M_{ef}\Big]\overline{y}\langle \mathrm{ElectronicFenceConf}\rangle \\ & \cdot \Big(!\Big(\mathrm{Int}(t_6, 30)\mathrm{Pos}\Big[S_{ef}\Big]\mathrm{Ene}\Big[M_{ef}\Big]\overline{z}\langle \mathrm{ElectronicFenceInf}\,A\rangle \cdot \mathrm{Int}(t_{Ae}, 30)\mathrm{Pos}\Big[S_{ef}\Big]\mathrm{Ene}\Big[M_{ef}\Big] \\ & \times z(\mathrm{Conf}\,A)\Big) \;|!\Big(\mathrm{Int}(t_6, 30)\mathrm{Pos}\Big[S_{ef}\Big]\mathrm{Ene}\Big[M_{ef}\Big]\overline{u}\,\langle \mathrm{ElectronicFenceInf}\,B\rangle \\ & \cdot \mathrm{Int}(t_{Be}, 30)\mathrm{Pos}\Big[S_{ef}\Big]\mathrm{Ene}\Big[M_{ef}\Big]u(\mathrm{Conf}\,B)\Big)\Big). \end{aligned} \tag{7}$$

(4) Early Warning and Accident Told. Since not directly interact with S, they are regarded as internal services.

Let P_{ag} denote CPS service "agent of traffic accident treatment platform" in Figure 6. Space constraints are $S_{ag} = \{S_{\text{pos}}, S_{\text{pos}}, \ldots S_{\text{pos}}\}$ and M_{ag}. Let P'_{rd}, P'_{ef} denote upgraded CPS services Remote Diagnosis and Electronic Fence, and other CPS services remain unchanged.

(1) Remote Diagnosis. Within $t_3 + 1$ of confirming as corresponding accident level, traffic accident information must be sent to agent of traffic accident treatment platform, and other flows remain unchanged.

$$\begin{aligned} P'_{rd} = {} & \mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]x(\mathrm{AlarmReq}) \\ & \cdot (\mathrm{Int}(t_2, 30)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]\overline{x}\langle \mathrm{AlarmConf}\rangle + \mathrm{Int}(t_2, 30)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]\overline{x}\langle \mathrm{Ignorance}\rangle) \\ & \cdot [\mathrm{msg} = \mathrm{AlarmConf}] \\ & \times (\mathrm{Int}(t_3, 1)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]\overline{v} \\ & \times \langle \mathrm{AccidentInf}\rangle \mid (\mathrm{Int}(t_3, 1)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]\overline{y}\langle \mathrm{ElectronicFenceReq}\rangle \\ & \cdot \mathrm{Int}(t_4, 60)\mathrm{Pos}[S_{rd}]\mathrm{Ene}[M_{rd}]y(\mathrm{ElectronicFenceConf}))). \end{aligned} \tag{8}$$

(2) Electronic Fence. Within $t_6 + 1$ of setting electronic fence, electronic fence information must be sent to agent of traffic accident treatment platform, and other flows remain unchanged

$$\begin{aligned}
P'_{ef} = {} & \mathrm{Pos}\left[S_{ef}\right]\mathrm{Ene}\left[M_{ef}\right]y(\mathrm{ElectronicFenceReq}) \\
& \cdot \mathrm{Int}(t_5, 30)\mathrm{Pos}\left[S_{ef}\right]\mathrm{Ene}\left[M_{ef}\right]\overline{y}\langle \mathrm{ElectronicFenceConf}\rangle \\
& \cdot \Big(\Big(\mathrm{Int}(t_6, 1)\mathrm{Pos}\left[S_{ef}\right]\mathrm{Ene}\left[M_{ef}\right]\overline{w}\langle \mathrm{ElectronicFenceInf}\rangle\Big) \ |! \\
& \times \Big(\mathrm{Int}(t_6, 30)\mathrm{Pos}\left[S_{ef}\right]\mathrm{Ene}\left[M_{ef}\right]\overline{z}\langle \mathrm{ElectronicFenceInf}A\rangle \\
& \cdot \mathrm{Int}(t_{Ae}, 30)\mathrm{Pos}\left[S_{ef}\right]\mathrm{Ene}\left[M_{ef}\right]z(\mathrm{Conf}A)\Big) \ |! \\
& \times \Big(\mathrm{Int}(t_6, 30)\mathrm{Pos}\left[S_{ef}\right]\mathrm{Ene}\left[M_{ef}\right]\overline{u}\langle \mathrm{ElectronicFenceInf}B\rangle \\
& \cdot \mathrm{Int}(t_{Be}, 30)\mathrm{Pos}\left[S_{ef}\right]\mathrm{Ene}\left[M_{ef}\right]u\,(\mathrm{Conf}B)\Big)\Big)
\end{aligned} \tag{9}$$

(3) Agent of Traffic Accident Treatment Platform.

$$\begin{aligned}
P_{ag} = {} & \mathrm{Pos}\left[S_{ag}\right]\mathrm{Ene}\left[M_{ag}\right]v(\mathrm{AccidentInf}) \\
& \cdot \mathrm{Pos}\left[S_{ag}\right]\mathrm{Ene}\left[M_{ag}\right]w(\mathrm{ElectronicFenceInf}).
\end{aligned} \tag{10}$$

From the results of above analysis, $P_S = P_{rd}$, $P_{S'} = P'_{rd} \mid P_{ag}$.

4.3. Analysis Results. In order to decide whether S' can be substituted for S, it is needed to determine if S' meets the four conditions in Theorem 12.

(1) For condition 1, taking software and hardware attributes of actual CPS services into $P_{S'}$ to deduce formally, then we can judge whether S' can reach final state. If applicable, S' meets condition 1.

(2) For condition 2, let $M = \{\text{Vehicle Alarm, Electronic Fence}\}$ denote set of all CPS services compatible with S. As can be seen from Figure 6, S' is static compatible with Vehicle Alarm and Electronic Fence, therefore S' meets condition 2.

(3) For condition 3, assuming S' meets Figure 6, then

$$\begin{aligned}
P_{\overline{S'}} = {} & \overline{P'_{rd} \mid P_{ag}} = \overline{x}\langle \mathrm{AlarmReq}\rangle \cdot (x(\mathrm{AlarmConf}) + x(\mathrm{Ignorance})) \cdot [\mathrm{msg} = \mathrm{AlarmConf}] \\
& \times (v(\mathrm{AccidentInf}) \mid (y(\mathrm{ElectronicFenceReq}) \cdot \overline{y}\langle \mathrm{ElectronicFenceConf}\rangle)) \\
& \mid (\overline{v}\langle \mathrm{AccidentInf}\rangle \cdot \overline{w}\langle \mathrm{ElectronicFenceInf}\rangle) \\
= {} & \overline{x}\langle \mathrm{AlarmReq}\rangle \cdot (x(\mathrm{AlarmConf}) + x(\mathrm{Ignorance})) \cdot [\mathrm{msg} = \mathrm{AlarmConf}]((y(\mathrm{ElectronicFenceReq}) \\
& \cdot \overline{y}\langle \mathrm{ElectronicFenceConf}\rangle) \mid \overline{w}\langle \mathrm{ElectronicFenceInf}\rangle).
\end{aligned} \tag{11}$$

We use MWB software tool to derive that $P_{\overline{S'}} \approx P_{va}$, $P_{\overline{S'}} \approx P_{ef}$, and for $P_M = \{P_{va}, P_{ef}\}$, S' meets condition 3.

(4) For condition 4, seen from Figures 5 and 6,

$$\begin{aligned}
& \mathrm{emissions}(S) \\
& \quad = \{\mathrm{AlarmConf}, \mathrm{Ignorance}, \mathrm{ElectronicFenceReq}\}; \\
& \mathrm{receptions}(S) = \{\mathrm{AlarmReq}, \mathrm{ElectronicFenceConf}\}; \\
& \mathrm{emissions}(S') \\
& \quad = \{\mathrm{AlarmConf}, \mathrm{Ignorance}, \mathrm{ElectronicFenceReq}\}; \\
& \mathrm{receptions}(S') \\
& \quad = \{\mathrm{AlarmReq}, \mathrm{ElectronicFenceConf}, \mathrm{ElectronicFenceInf}\}.
\end{aligned} \tag{12}$$

Obviously, emissions (S) $\subseteq$ emissions (S'), receptions $(S) \subseteq$ receptions (S'), so S' meets condition 4.

From the above, as long as S' meets condition 1, S' can be substituted for S directly.

As can be seen from Figure 6, there are five normal interaction elements and one abnormal interaction element in S'. According to Definition 3, $N(IE_n(S')) = 5$, $N(IE_a(S')) = 1$, $\omega = N(IE_n(S'))/(N(IE_n(S')) + N(IE_a(S'))) = 5/6$. By the substitution processes mentioned in Section 3.3, problem management process and change management process are adopted to minimize the effects of abnormal interaction elements and ensure the substitution reliability.

In actual operation, after three rounds testing and half year's test running, substituted electronic fence runs well, which fully proves that the above analysis results are correct. This case study shows that the reliability assurance method mentioned in the paper can assist users in CPS components substitution and ensure the reliability of upgraded CPS; therefore, this method is reasonable and feasible.

5. Conclusions

In this paper, CPS components substitution is equated to CPS service substitution, and a reliability assurance method for CPS service substitution is provided. The case study proves that the method is innovative and practical. Our future works will focus on two aspects: (1) how to realize incompatible CPS service substitution through adding process adapter, so as to expand the sample selection space. (2) Take further study on action-time function and action-energy function, construct time and energy state space, then we can make optimal service composition decision in this state space, and provide reference for the optimization selection of CPS service substitution.

Acknowledgments

The authors would like to respectively thank Dr. Xing Zhao and Dr. Liwei Sheng from College of Computer Science and Technology, Fudan University for providing useful insights. This work is supported by National High Technology Research and Development Program of China (no. 2011AA010101), National Natural Science Foundation of China (no. 71171148), and key research project of Shanghai Science and Technology Committee (no. 11DZ1501703).

References

[1] E. A. Lee, "Cyber physical systems: design challenges," in *Proceedings of the 11th IEEE Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC '08)*, pp. 363–369, Orlando, Fla, USA, May 2008.

[2] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: the next computing revolution," in *Proceedings of the 47th Design Automation Conference (DAC '10)*, pp. 731–736, New York, NY, USA, June 2010.

[3] Y. Tan, S. Goddard, L. C. Perez et al., "A prototype architecture for cyber-physical systems," *ACM SIGBED Review*, vol. 5, no. 1, pp. 56–60, 2008.

[4] L. T. X. Phan and I. Lee, "Towards a compositional multi-modal framework for adaptive cyber-physical systems," in *Proceedings of the 17th International Conference on Embedded and Real-Time Computing Systems and Applications*, pp. 67–73, Toyama, Japan, 2011.

[5] A. Koubaa and B. Andersson, "A vision of cyber-physical internet," in *Proceedings of the 8th International Workshop on Real-Time Networks*, pp. 75–80, Porto, Portugal, 2009.

[6] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*, Prentice Hall PTR, Upper Saddle River, NJ, USA, 2005.

[7] R. A. Thacker, K. R. Jones, C. J. Myers, and H. Zheng, "Automatic abstraction for verification of cyber-physical systems," in *Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS '10)*, pp. 12–21, New York, NY, USA, April 2010.

[8] E. A. Lee and S. Tripakis, "Modal models in ptolemy," in *Proceedings of the 3rd International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools*, pp. 11–21, Oslo, Norway, 2010.

[9] R. Akella and B. M. McMillin, "Model-checking BNDC properties in Cyber-physical systems," in *Proceedings of the 33rd Annual IEEE International Computer Software and Applications Conference (COMPSAC '09)*, pp. 660–663, Seattle, Wash, USA, July 2009.

[10] R. Milner, *Communicating and Mobile Systems: The π-Calculus*, Cambridge University Press, 1999.

[11] L. Bordeaux, G. Salaün, D. Berardi, and M. Mecella, "When are two web services compatible?" in *Proceedings of the 5th International Workshop on Technologies for E-Services (TES '04)*, pp. 15–28, Toronto, Canada, August 2004.

[12] M. Iqbal and M. Nieves, *ITIL Service Strategy*, The Stationery Office, London, UK, 2007.

[13] ISO/IEC 20000-1: 2005, *Information Technology Service Management-Part 1: Specification*, International Organization for Standardization, 2005.

[14] J. M. M. Perez, J. B. Bernabe, D. J. M. Manzano, G. M. Perez, and A. F. G. Skarmeta, "Towards the definition of a web service based management framework," in *Proceedings of the 2nd International Conference on Emerging Security Information, Systems and Technologies*, pp. 362–367, Cap Esterel, 2008.

[15] V. Machiraju, A. Sahai, and A. V. Moorsel, "Web services management network: an overlay network for federated service management," in *Proceedings of the 8th International Symposium on Integrated Network Managemen*, pp. 351–364, Ghent, Belgium, 2003.

[16] OPEN GIS CONSORTIUM, "OpenGIS geography markup language(GML) encoding standard," Version 3. 3, 2007.

[17] M. J. Egenhofer and J. R. Herring, "A mathematical framework for the definition of topological relationships," in *Proceedings of the 4th International Symposium on Spatial Data Handling*, pp. 803–813, Zurich, Switzerland, 1990.

[18] V. Bjorn and M. Faron, "The mobility workbench-a tool for the π-calculus," *Computer Aided Verification*, vol. 818, pp. 428–440, 1994.

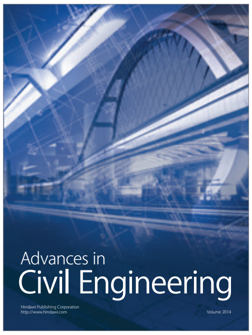
Journal of
Engineering

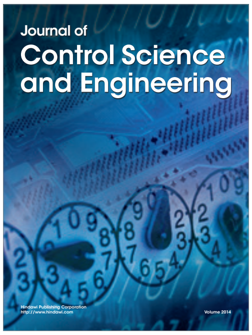
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

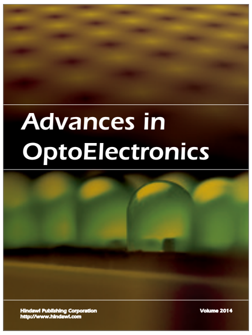
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

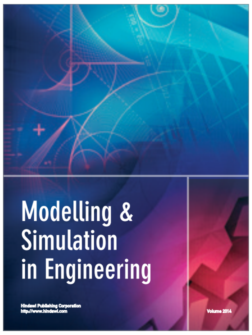
Journal of
Robotics

Journal of
Electrical and Computer Engineering

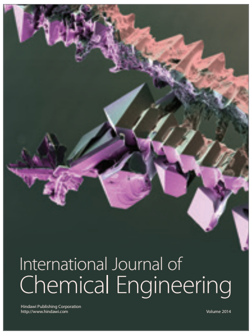

Advances in
OptoElectronics

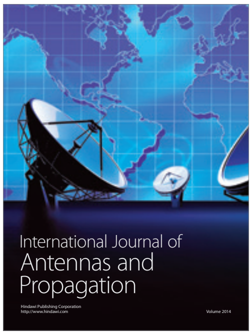
VLSI Design

International Journal of
Navigation and Observation

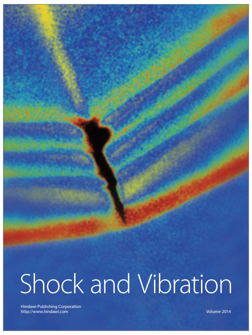

Modelling & Simulation in Engineering

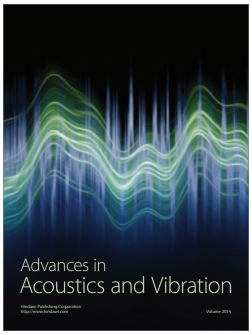
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration