Modeling and Analyzing Reliable Cyber-Physical Systems Based on Aspect Orientation

Liqiong Chen\textsuperscript{1,2}, Yunxiang Liu\textsuperscript{1} and Guisheng Fan\textsuperscript{3}

\textsuperscript{1}Department of Computer Science and Information Engineering, Shanghai Institute of Technology, Shanghai, P.R. China
\textsuperscript{2}Shanghai Key Laboratory of Computer Software Evaluating and Testing, Shanghai, P.R. China
\textsuperscript{3}Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai, P.R. China

Abstract

Cyber-physical systems (CPS) consists of distributed computation interconnected by computer networks. Finding a common semantics among these diverse components that facilitate system modeling, synthesis, and verification is a significant challenge of CPS. Aspect oriented method is proposed to model and analyze reliable CPS, Petri nets are used to observe behaviors of basic component, such as device, computation and physical interaction, component and communication process. The reliability assurance strategy of CPS is proposed, aspect orientation is used to weave it into the base net. Based on this, the effectiveness and feasibility of proposed method are analyzed based on the operation semantics of Petri nets. A case study of material balancing system and experimental results demonstrate that the approach can contribute to the improvement of design quality in developing reliable CPS.

Key Words: Cyber Physical Systems, Petri Nets, Communication, Aspect Orientation, Reliability

1. Introduction

Cyber-physical systems (CPS) involves communication, computation, sensing, and actuating through heterogeneous and widely distributed physical devices and computational components \cite{1}. The economic and societal potential of CPS is believed to be tremendously greater than what has been achieved by existing systems in terms of autonomy, flexibility, and versatility \cite{2}. However, as CPS is geographically distributed without centralized control, no guarantees can be made about the reliability of the resulting system. And CPS includes hardware, software, and physical environment, which makes the problem of reliability more challenging than conventional client-server systems.

Aspect-oriented programming (AOP) \cite{3} is a software engineering approach to separation of concerns, or the ability to specify, encapsulate, and manipulate only the parts of software. The significance of aspect oriented design (AOD) at more abstract level has come to front recently \cite{4}, because quality assurance in the early stage of software life cycle can result in better design, and shorter time to market. However, AOP is mainly used in the coding stage, and the requirements of aspect-oriented analysis often appear ambiguous and vague. Therefore, formal methods can be used to analyze AOP, thus increasing the semantic constraints. Petri net is a mathematically based technique for modeling and verifying software artifacts \cite{5}. It provides clear and precise formal semantics, an intuitive graphical notation, and many techniques and tools for their analysis and simulation.

In this paper, aspect oriented approach to modeling and analyzing reliable CPS is proposed, Petri net is used to model device, computational and physical interaction,
component and communication process. The reliability assurance strategy for CPS is proposed, aspect specification provides means to observe behaviors of basic aspect schema, such as device failure concern, task failure concern and communication failure concern, the weaving mechanism dynamically integrates these schemas into a reliability model. The operational semantics and related theories of Petri net help prove the correctness of proposed method at design time, and to analyze the reliability of CPS at runtime according to the actual requirement. A case study of material balancing system and experimental results demonstrate that the proposed method can not only contribute to the improvement of design quality, but also inline with the current research trend.

The remainder of this paper is organized as follows: Section 2 is the requirements of CPS. In section 3, we present the composition model, the basic definition is also given. Section 4 presents the analysis techniques. In section 5, we explain the effectiveness of our methods by a specific example. Section 6 presents some related works while section 7 is conclusion.

### 2. Requirements of CPS

The structure of CPS is composed of computational, physical and hybrid system. Computational system is used to support the real-time monitoring and data processing. Physical system is used to process the underlying device. Hybrid system is the connection between computational system and physical system. Each sub-system contains the component, connector and its attributes. The component is used to realize the functions of CPS, such as resource invocation, computation and storage, etc. The connector mainly realizes the communication between components. Each component, connector can be divided into a series of interrelated tasks. In order to protect the sensitive data, the access of task must meet certain conditions.

Publish/subscribe system is suitable for CPS [6], which has the dynamic, autonomous and loosely coupled characteristics. The bus of publish/subscribe system is used to realize the communication between components.

**Definition 1:** The requirement of CPS is 6-tuple: \( \Xi = (\Delta e, C, HC, Bri, DC, RD) \):

1. \( \Delta e = (DeS, DeA, DeN) \) is the set of three type for devices, \( DeS, DeA, DeC \) are the sensor, actuator and network node respectively.
2. \( C = (\text{sub}C, R, RN, TE, RL, RM) \) is the set of component, each component has six attributes, where: \( \text{sub}C = TK \lor \text{SC} \) is the set of task or sub-component. \( R \) is the set of resource. \( tk_{ij}, sc_{ij}, ri_{ij} \) are the \( j \)th task, sub-component and resource of \( C \) respectively. \( RN: R \rightarrow N^* \) is the number of resource of component. \( TE \) is the attributes of task. \( RL: \text{sub}C \times \text{sub}C \rightarrow \{ >, \|, +, \oplus \} \), \( \|, +, \oplus \) represent the sequence, choice, parallel and exclusive relationship respectively. \( RM \) is the communication between tasks. The task that needs to publish event or subscribe event in component \( C_i \) is denoted by set \( Pmt(C_i) \) and \( Smt(C_i) \).
3. \( HC: C \rightarrow \{ C, Py, H \} \) is the attribute function of component.
4. \( Bri \) is a list of route, which describes the transmission path of CPS.
5. \( DC: C \rightarrow De^* \) is used to allocate the device for component in the physical system.
6. \( RD: DeS \cup DeA \rightarrow (0, 1) \) is the reliability of sensor and actuator.

In this paper, we assume that CPS has the following characteristics. Priority is used to compete for resource. The reliability of communication is \( r \). The same type of unavailable device (sensor, actuator) in the component is not more than \( \hat{c} \) (which can be set according to the actual requirements) at the same time. The time that device restores to the normal state from the failure state is \( Tr \), backup device cannot fail at this stage.

### 3. Computation Model

#### 3.1 Syntax and Semantics

The function of CPS is composed by a number of components. The composition net is used to describe the execution process of CPS. Advice net and introduction net correspond to the pointcut of cross-cutting concerns. The weaving of composition net, advice net and introduction net of crosscutting concern according to a certain weaving rules will form an aspect net.

**Definition 2:** An 8-tuple \( \Sigma = (N, h, IO, D, A_T, A_F, \lambda, \) ...
$M_0$ is called a base net (BN), where:

1. $N=(P,T,F)$ is a Petri net. $P$, $T$, $F$ are the place, transition and arc respectively.
2. $h: P \cap T \rightarrow \{D, C\}$ is a hybrid function, which is used to divide the place and transition into discrete node ($P^D$ and $T^D$) and continuous node ($P^C$ and $T^C$).
3. $IO \subset P$ is a special type of place, which is the interface of $\Sigma$ and denoted by dotted circle.
4. $D$ is a non-empty set of individuals, which decides the type, operation and function of marking in $\Sigma$. $f_D$, $f_S$ are the set of formula (Boolean expression) and symbol.
5. $A_T: T \rightarrow f_D$, $t \in T$, $A_T(t)$ contains the static predicate and operator in $D$.
6. $A_F: F \rightarrow f_S \times R_0^+$. $\lambda: T \rightarrow N^+ \times R_0^+ \times (0,1)$ is the attribute function of transition. $R_0^+$ is the real number. $\lambda$ describes the priority, delay time, and firing probability of transition.
7. $M_0: P \rightarrow f_S \times R_0^+$ is the initial marking of $\Sigma$.

Individual set $D$ is mainly used to describe the resources, such as resource and data packet. The task and component can control the operation of CPS by using the message. The data packet of message in CPS is abstracted as an individual $\varepsilon$. The individual in the model is $\varepsilon$. For all $x \in (P \cup T)$, we denote $x(t^*,n)$ as the $n$ pre-set of $x$, the input/output arc of transition $t_i$ and the free variable in $A_T(t_i)$ are denoted by $FV(t_i)$.

**Definition 3**: A six-tuple $\Omega=(\Sigma, \Gamma, TI, TA, PI, PA)$ is called composition net (CN):

1. $\Sigma$ is a BN model, which describes the basic structure of $\Omega$.
2. $\Gamma = \{\Gamma_i|i \in N^+\}$ is a finite set of page. Each page is a BN or CN model, and it is disjoint between pages.
3. $TI \subset T, PI \subset P$ are the set of substituted node and interface node.
4. $TA: TI \rightarrow \Gamma$ is the page allocation function.
5. $PA$ is the mapping function of interface.

CN is mainly used to model the device, control process and host. In order to ensure the non-negative of $P^D$.

$\forall p \in P^D, \forall t \in T^C$, if $(p,t) \in F$, then $(t,p) \in F \wedge A_T(p,t) = A_F(t,p), v_i(\tau)$ is the instantaneous firing probability of transition $t_i$ at time $\tau$. Place $p$ only contains an individual.

In order to distinguish the transition and place in each net, the element $x$ in net $N_i$ is denoted by $N_i, x_i$, and the incidence matrix of composition net $\Omega$ is $IM(\Omega)$.

The mapping $CutN: \{N_1, x_j, N_m, x_m, \ldots, N_f, x_l\}$ is called a pointcut, $CutN$ is the name of pointcut, $N_i, x_j$, $N_m, x_m$, $\ldots$, $N_f, x_l$ are the joinpoint of pointcut. A triple $con = (CutN, AN, IN)$ is called a concern, $CutN, AN, IN$ represent the pointcut, advice net and introduction net of the concern respectively.

Let $\Omega$ be the model before weaving, $con$ is the required weaving pointcut, then the steps for constructing the aspect net $\Omega'$ are as follows:

1. Adding the introduction net to $\Omega$ at the pointcut based on its definition.
2. Adding the element and merging the same element according to the weaving rules.

The distribution of individual in each place at time $\tau$ is called the marking of CN, denoted by $M^\tau$. The replacement is mainly used to bind the individual to the input/output arc of $t$ and all free variables in $A_T(t)$. $t<d_1, d_2, \ldots, d_\circ >$ and $A_F(p,t)<d_1, d_2, \ldots, d_\circ >$ are the values got by replacing formulas $A_T(t)$ and the predicates $A_F(p,t)$ of input arc with individuals $d_1, d_2, \ldots, d_\circ$. If $t<d_1, d_2, \ldots, d_\circ >$ which makes $A_T(t)<d_1, d_2, \ldots, d_\circ >$ true, then $t<d_1, d_2, \ldots, d_\circ >$ is called a feasible replacement of transition $t$ under marking $M^\tau$. All feasible replacements of transition $t$ under marking $M^\tau$ are denoted by set $VP(M^\tau, t)$.

**Definition 4**: Let $\Omega$ be a CN model, $M^\tau$ is a marking of $\Omega$ at time $\tau$. $\forall t \in T, if VP(M^\tau, t) \neq \emptyset, and \exists !<d_1, d_2, \ldots, d_\circ > \in VP(M^\tau, t)$, which makes:

1. $If t \in T^C: (\forall p \in t^*, M^\tau(p) \geq A_F(p,t) <d_1, d_2, \ldots, d_\circ >)$.
2. $If t \in T^C: (\exists p \in t^* \wedge P^D, M^\tau(p) \geq A_F(p,t) <d_1, d_2, \ldots, d_\circ > \wedge (\forall p \in t^* \wedge P^D, M^\tau(p) \neq \emptyset \vee (\exists t_i \in T^C \wedge p, v_i(\tau) > 0 \wedge p \notin t_i))$.

Then transition $t$ is enabled under marking $M^\tau$. All the enabled transitions under $M^\tau$ are denoted by set $ET(M^\tau)$. The firing of transition $t$ is effective under marking $M^\tau$ if and only if transition $t$ is enabled and $ET(M^\tau)$ doesn’t have the transition whose priority is greater than $t$. All effective firing transitions under marking $ET(M^\tau)$ are denoted by set $FT(M^\tau)$. The process that marking $M$ reaches a new marking $M'$ by firing a feasible replacement $t<d_1, d_2, \ldots, d_\circ >$ of transition $t$ is denoted by $M(t<d_1, d_2, \ldots, d_\circ > M')$. The concurrent transitions under marking $M^\tau$
are denoted by set $MT(M^*)$. The set $H(M^*) = \{ t < d_1, d_2, \ldots, d_n, t \in M(T(M), t < d_1, d_2, \ldots, d_n) \in VP(M, t) \}$ is called the greatest concurrent set of $M^*$.

**Definition 5**: Let $M^*$ be a marking of $\Omega$ at time $\tau$. The model will reach a new state $M^{t+\omega}$ by effectively firing all enabled transitions in $H(M^*)$ at time $\tau + \omega (\omega > 0)$, denoted by $M[H(M^*)] > M^{t+\omega}$. $M^{t+\omega}$ is called the reachable marking, which is computed based on the following rules:

1. If $t_i \in T^D: \forall p_j \in t_i \cap T^*; M^{t+\omega}(p_j) = M^1(p_j) - A_F(p_j, t_i) < d_1, d_2, \ldots, d_n > + A_F(t_i, p_j) < d_1, d_2, \ldots, d_n >$.

2. If $t_i \in T^C: M^{t+\omega}(p_j) = M^1(p_j) - \sum_{i = 1}^{\infty} V_i(\tau) d\tau$.

Vector $S(M^*) = [s_i]$, each element $s_i$ equate as: $s_i = \begin{cases} 1, & t_i \in FT(M^*) \cap T^D \\ \{v_j(\tau), t_i \in FT(M^*) \cap T^C. \end{cases}$ Then Definition 5 can be converted into: $M^* = M^1 + IM(\Omega)$-$S(M^*)$. Therefore, we can use the incidence matrix to analyze CN model. If there is a firing sequence $H_1, H_2, \ldots, H_n$ and state sequence $M_1, M_2, \ldots, M_n$ which make $M[H_1] > M[H_2] > M_2 \ldots M_n], M^1 > M_n$, then $M_n$ is a reachable state from $M$. All the possibly reachable states of $M$ are denoted by $R(M)$ and $M \in R(M)$. All firing sequences from $M$ to $M_n$ are denoted by set $\delta(M, M_n)$.

### 3.2 Core Net of CPS

In this section, we will model different components of CPS.

**Modeling device**: The CN model of sensor $de_i$ is shown in Figure 1(a), the specific process is: if the sensor is invoked, then fire transition $t_{in}$ to initialize the device. The delay time $\beta(t_{in})$ of $t_{in}$ is equal to the sampling time interval of sensor. The CN model of actuator $de_j$ is shown in Figure 1(b), $t_{adj}$ is the adjustment operation. The priority of transition $t_{adj}$ is higher than $t_{adj}$.

**Modeling the interaction between computational system and physical system**: The CN model of the interaction between physical system and computational system is shown in Figure 2. Figure 2(a) describes the process that computational system affects the physical system. Let $M(p_2) = e, [M(p_2)] \neq \emptyset$. Figure 2(b) is used to describe the process that physical system affects the computational system. Let $M(p_1) = e, [M(p_1)] \neq \emptyset$. Figure 2(c) and (d) are used to describe the process that the continuous marking transfers into the discrete marking and its reverse process.

**Modeling component**: The CN model of component $Ci$ is $CN_i$, the specific process is:

1. Modeling task and sub-component.
2. Modeling resource: Place $p_i$ is used to represent the distribution of shared resources in component, while the resource $r$ is abstracted as $d_r$. Let the number of $r$ be $n$, then $M_0(p_i) = (d_r, n)$ is used to represent that there is $n$-such available resources.
3. Modeling the basic relation between tasks.
4. Introducing $t_{in}$ and $p_i$ (input interface) to describe the beginning operation and position, and initializing the whole system based on the characteristics of component. $t_{in}$ and place $p_i$ are introduced to describe the operation and position of the whole system.
5. Composing the above model according to the basic relations between task and sub-component. The priority of transition is set to 3 (the lowest level).

**Modeling the communication process**: We abstract...
the event as \( d_i \) (event number, event type, event publisher and access condition), the event number is sole. The CN model of publishing and subscribing event between components is shown in Figure 3 (a)–(b). We can assume that task \( t_k,i \), \( t_k,g \), \( \ldots \), \( t_k,g \) are the producer, their execution will publish the event. Task \( t_k,i \) is a consumer.

According to the characteristics of CPS, the core net \( \Omega \) is constructed by using the following steps:

1. Constructing the model of device and component according to their modeling step.
2. Composing the model of component according to their communication process.
3. Introducing \( t_a, t_r, p_a, \) and \( p_t \) to describe the beginning, termination operation and position of the whole system based on the characteristics of component.

### 3.3 Basic Properties of Constructed Model

Let the CN model of component \( C_i \), sub-component \( sc_{i,j} \), device \( d_e \) be \( CN_i, scNi_j, deNi \), the transition and place in model \( scNi_j \) are denoted by set \( T_{sj} \) and \( P_{sj} \). \( M^1 \) is the marking of component \( C \) at time \( \tau \), then: (1) The individuals in the places of model \( scNi_j \) are denoted by \( TM(M^1, sc_{i,j}) \), which is called the mapping of sub-component \( sc_{i,j} \) on \( M^1 \). (2) \( \forall M \in R(M^1) \), \( \delta(M^1, M) \) is the firing sequence from \( M^1 \) to \( M \). (3) \( \forall \delta^k \in \delta(M^1, M) \), the firing sequence \( \delta_{k}^w = \{ \delta_t | \delta_t \in \delta^k \cap T_{i,j} \} \) is called the project sequence of \( \delta^k \) on sub-component \( sc_{i,j} \).

If \( M(p_a) = e \), then \( M \) is called a normal termination marking. The normal termination marking of \( \Omega \) is denoted by set \( M^\delta(\Omega) \).

**Theorem 1:** Let \( M^1 \) be a marking of component \( C \) at time \( \tau \), \( TM(M^1, sc_{i,j}) \) is the mapping of marking \( M^1 \) on the sub-components \( sc_{i,j} \), then \( \forall M \in R(M^1) \), \( \forall \delta^k \in \delta(M^1, M) \), there is \( TM(M^1, sc_{i,j}) \{ \delta_{k}^w \} \geq TM(M^1, sc_{i,j}) \).

Proof: Because \( TM(M^1, sc_{i,j}) \) is the mapping of \( M^1 \) on \( sc_{i,j} \). According to the definition of function \( TM \), \( \forall \delta_t \in P_{i,j} \), there is \( M^1(p_a) = TM(M^1, sc_{i,j}) \).

Therefore, \( TM(M^1, sc_{i,j}) \) is a marking of \( sc_{i,j} \), we can randomly select a greatest concurrent set \( H_1(M^1) \) from \( M^1 \).

Because \( sc_{i,j} \) is a sub-component of \( C_i \), we can get: \( \forall t \in T_{i,j}, \{ \tau \cap \tau \} \cap (P_i \cap P_{i,j}) = \emptyset \).

Therefore, the firing of transition \( t_i \) is only related with the individuals in \( P_{i,j} \), that is, there is a greatest firing sequence \( H_2(TM(M^1, sc_{i,j})) \) of \( TM(M^1, sc_{i,j}) \), which makes \( H_2(TM(M^1, sc_{i,j})) \), so \( TM(M^1, sc_{i,j}) \{ H_2(TM(M^1, sc_{i,j})) \} \geq TM(M^1, sc_{i,j}) \).

Because \( H_2(TM(M^1, sc_{i,j})) = H_1(M^1) \cap T_{i,j} \). Therefore, \( M_1 = TM(M_1, sc_{i,j}) \), and we can get: \( \forall M^1 \in R(M^1), \forall \delta^k \in \delta(M^1, M^1), \delta_{k}^w \geq TM(M^1, sc_{i,j}) \).

In summary, let \( M^1 \) be a marking of \( C_i \) at \( \tau \), \( TM(M^1, sc_{i,j}) \) is the mapping of \( M^1 \) on the \( sc_{i,j} \), then \( \forall M^1 \in R(M^1) \), \( \forall \delta^k \in \delta(M^1, M^1) \), there is \( TM(M^1, sc_{i,j}) \{ \delta_{k}^w \} \geq TM(M^1, sc_{i,j}) \).

Theorem 1 explains that we can divide the CN model of CPS into a series of sub-components according to the actual requirements.

### 4. Reliability Assurance Techniques of CPS

#### 4.1 Reliability Assurance Strategy of CPS

We will propose the reliability assurance strategy for CPS according to the different failure type (We can assume that the required reliability is \( DP \), which includes:

**Device strategy:** (1) If the sensor has the computation model, then the system will firstly use the simulation model to predict the collected data of sensor and compare it with the actual data, once the difference is greater than the threshold \( t \) or the sensor cannot collect the data, then the sensor fails or the actuator becomes unavailable. (2) The system will add \( \delta \) backup devices to each component \( C \). Once the failure info of device is detected, the system will invoke an available backup device.

If the sensor restores to the normal state, then interrupt the operation of backup sensor and reinvoke the original sensor. If the actuator operates normally, then it will be in waiting for the next invocation.

**Task strategy:** The redundant technology is used to do the redundant computation for the critical task.

![Figure 3. The CSPN model of communication process.](image-url)
(1) As the task in CPS has the high reliability, we mainly use two or three backup devices. (2) All backup devices will operate in parallel and amend their own state once they receive the request, then return the results. (3) If all backup devices fail, it explains that the execution of task fails.

Let the corn net of CPS be \( \Omega \) and \( M_k \in M(\Omega) \), the set \( E(T_k) = \{ t_{i,j} | i,j \in \delta(M,M_k) \} \) is called a complete task set. All the complete tasks are denoted by set \( AET(\Omega) \).

Assume that the packet data in communication strategy can be sent \( y \) times. We will give the steps to compute the number of backup device BackN\((tk_{i,j}) \) of task \( tk_{i,j}(\text{BackN}(tk_{i,j})) = 0 \), and \( AE = AET(\Omega) \):

If \( AE \neq \emptyset \), then \( \forall E(T_k) \in AE, \forall tk_{i,j} \in ET_k \), and do the following step: Computing the threshold \( z, N_p = [ET_k \cap Sm(\Omega)] \), then \( z = \frac{DP}{(1-(1-r)^{\gamma})^{ET_k}} \). The reliability \( rt_{i,j} \) of task: \( \text{BackN}(tk_{i,j}) = \max \{ \text{Round}(\log_{1-(1-r)} z), \text{BackN}(tk_{i,j}) \} \), \( AE = AE - ET_k \), \( \text{Round}(x) \) is a rounded function for \( x \).

According to the attributes of task and its backup device, we will compute the reliability of complete task set: \( \forall ET_k \in AET(\Omega) \), where the success probability of \( ET_k \) is equal to: \( FR(ET_k) = \prod_{t_{i,j} \in ET_k} (1-(1-r)^{\gamma})^{\text{BackN}(tk_{i,j})} \times \prod_{t_{i,j} \in ET_k \cap Sm(\Omega)} (1-(1-r)^{\gamma}) \).

**Communication strategy:** The system will re-select the resource when the route node fails. Each data packet can only be sent \( y \) times (\( y = \text{Round}(\log_{1-(1-r)} \frac{DP}{(1-(1-r)^{\gamma})^{ET_k}}) \)). \( X \) is the number of task and the request, if any data packet fails, the system will compete the route again.

The communication will fail when it has operated \( y \) times. We will use CN to model failure outputting concern \( con_{sf} \), device failure concern \( con_{df} \), task failure concern \( con_{df} \) and communication failure concern \( con_{sf} \). The model got by weaving \( \{ con_{df}, con_{df}, con_{df}, con_{df} \} \) into the core net is called the reliability model \( \Omega_r \).

(1) Failure outputting concern \( con_{sf} \)

First, we will give the definition of cutpoint: \( t_{sf} \) \( \{ \text{de} | \text{pr} \} \text{de} \in \text{DeS} \}, \{ \text{de} | \text{pr} \} \text{de} \in \text{DeS} \cup \text{DeA} \} \cup \{ \text{CN} | \text{pr} \} tk_{i,j} \in TK \} \cup \{ \text{CN} | \text{pr} \} tk_{i,j} \in Sm(\text{CN}) \} \). The introduction net is shown in Figure 4(a)-(b).

The cutpoint \( t_{sf} \) is used to output the maximum error of sensor data. We introduce \( p_{ae}, p_{te} \) and \( p_{ow} \) to describe the changing process of resource. The firing probability of \( t_{ae} \) is equal to the changed probability of resource. The weaving rule of \( con_{sf} \) is: for each task, device, and subscription request, the system will introduce the cutpoint of failure outputting concern and its introduction net. The priority of transition is set to 0.

(2) Device failure concern \( con_{df} \)

The definition of device failure pointcut is: \( p_{df} \) \( \{ \text{de} | \text{pr} \} \text{de} \in \text{DeS} \}, \{ \text{de} | \text{pr} \} \text{de} \in \text{DeS} \cup \text{DeA} \} \cup \{ \text{CN} | \text{pr} \} tk_{i,j} \in TK \} \cup \{ \text{CN} | \text{pr} \} tk_{i,j} \in Sm(\text{CN}) \} \). The introduction net is shown in Figure 4(c)-(f). \( p_{df} \) are used to describe the execution of backup sensor and re-start the sensor, \( p_{ow} \) describes the invocation of backup sensor, \( p_{df} \), \( p_{df} \), \( p_{df} \) are used to describe the execution of backup actuator and re-start the backup actuator.

(3) Task failure concern \( con_{tf} \)

The definition is: \( t_{tf} \) \( \{ \text{CN} | \text{pr} \} tk_{i,j} \in \text{sub} \} \), \( p_{tf} \) \( \{ \text{CN} | \text{pr} \} tk_{i,j} \in \text{sub} \} \), the introduction net is shown in Figure 5(a)-(b). \( t_{tf} \) describes the operation of task, for each backup device \( tk_{i,j} \), \( p_{tf} \) are used to describe the waiting for operation position, operation failure and result output. The weaving rule of \( con_{tf} \) is: the system will weave \( t_{tf} \) and \( p_{tf} \) into \( tk_{i,j} \) according to the actual requirements and characteristics of the component \(Ci\); the priority of introduced transitions is set to 0.

(4) Communication failure concern \( con_{cf} \)

The definition is: \( p_{cf} \) \( \{ \text{CN} | \text{pr} \} tk_{i,j} \in \text{Sm}(\text{CN}) \} \), the
introduction net is shown in Figure 5(c). If the communication fails ($t_i$) and can try again, then the system will be invoked again. If the number of repetition is $y$, then output the fail info. The weaving rule is: the system will name the introduced transition and place according to the actual requirements, and the priority of $t_i$ is equal to the success probability of communication.

The model got by weaving the above four concerns into the corn net is called a reliability model.

### 4.2 Feasibility and Effectiveness of Strategy

Let $M$ be a reachable marking of $\Omega_s$. If $FT(M) = \emptyset$, then $M$ is called a termination marking of $\Omega_s$. $TS(\Omega_s) = \{M | M \in R(\Omega_s) \land FT(M) = \emptyset\}$. If $M \in TS(\Omega_s) \land M(p_i) = e_i$, then $M$ is called a normal termination marking of $\Omega_s$. $TS^s(\Omega_s) = \{M | M \in TS(\Omega_s) \land M(p_i) = e_i\}$. $TS(\Omega_s) = TS(\Omega_s) - TS^s(\Omega_s)$. We will analyze the effectiveness of reliability assurance strategy.

**Theorem 2:** Let the reliability model be $\Omega_s$, $\exists tk_{i,j} \in TK$, $\Omega_s$ is got by weaving the task failure concern into $tk_{i,j}$. Let $M_0$ be the initial marking of $\Omega_s$, then:

1. $\forall M \in TS^s(\Omega_s)$, $\forall \delta^k \in \delta(M_0, M)$, $\forall C_i \in C$, $CN_i, tfa \not\in \delta^k$.
2. $\forall M \in R(\Omega_s)$, $\forall \delta^t \in \delta(M_0, M)$, $\exists \delta t_{i,j,k} \in D_{CK}, deN_i, tfa \not\in \delta^t$, then $\exists M' \in TS^s(\Omega_s)$, which makes $M' \in R(M)$.
3. $\forall M \in R(\Omega_s)$, $\forall \delta^t \in \delta(M_0, M)$, if $\exists t_{i,j,k} \in D_{CK}, t_{i,j,k} \not\in \delta^t$, then $\exists M' \in TS^s(\Omega_s)$, which makes $M' \in R(M)$.

Proof by contradiction: (1) Let the marking from $M_0$ to $M$ be set $rm = \{M_1, M_2, \ldots, M_r\}$. And $\forall M \in TS^s(\Sigma)$, $\forall \delta^t \in \delta(M_0, M)$, then $\exists C_i \in C$, $CN_i, tfa \not\in \delta^t$. Because $CN_i, tfa \in \delta^t$, we can get $\exists M' \in rm$, which makes $M'_s(CN_i, p_{fa}) = e$, and $C_j \in C$, $C_j \in sub C_j$. Because $M'_s(CN_i, p_{fa}) = e$, we can get $M'_s(CN_i, p_{fa}) = \emptyset$.

According to Theorem 1, $\forall M' \in R(M'_s)$, $M'_s(CN_i, p_{fa}) = \emptyset$, until $M'(p_{fa}) = \emptyset$. So, $\forall M' \in R(M'_s), M' \not\in TS^s(\Omega_s)$.

Because $M \in R(M'_s)$, we can get $M \not\in TS^s(\Omega_s)$, which is contradicted with the supposition.

So the supposition does not establish. We can prove sub-proposition (2) and (3) in the similar way.

### 5. Example

In this paper, we use a material balancing system as an example to illustrate the modeling and analysis process. $C_1$ is used to perform the oil distillation and produce kerosene. $C_2$ performs the gas oil industrial cracking, while $C_3$ is used to perform residue coking industry. $C_4$ tracks the storage of individual tank. $C_5$ accurately tracks the crude oil, $C_6$ includes production cost computation ($C_7$) and allocation of production ($C_8$). Where $C_1, C_2, C_3$ are sub-components of $C_4$, and $C_6, C_7, C_8$ are the component of computational system, the remaining components are the physical system. $C_1$: $tk_{1,1} > tk_{1,2} > (tk_{1,3} || tk_{1,4} || tk_{1,5} || tk_{1,6} || tk_{1,7} || tk_{1,8})$. $C_2$: $tk_{2,1} > tk_{2,2} > tk_{2,3}, C_3$: $tk_{3,1} > tk_{3,2} > (tk_{3,3} || tk_{3,4}), C_4$: $tk_{4,1} > (tk_{4,2} || tk_{4,3} || tk_{4,4} || tk_{4,5} || tk_{4,6} || tk_{4,7} || tk_{4,8} || tk_{4,9} \lor tk_{4,10} \lor tk_{4,11} > tk_{4,12}, C_5$: $(C_4 \lor C_5 \lor C_6) > tk_{5,1} > tk_{5,2} > (tk_{5,3} + tk_{5,4} > (tk_{5,5} || tk_{5,6} > tk_{5,7}, C_6$: $tk_{6,1} > C_7 > C_8 > tk_{6,2}, C_7$: $tk_{7,1} > tk_{7,2} > tk_{7,3}, C_8$: $tk_{8,1} > tk_{8,2} > tk_{8,3} > tk_{8,4} || tk_{8,5} || tk_{8,6} \lor tk_{8,7} || tk_{8,8} || tk_{8,9} || tk_{8,10} = tk_{8,11}$). The communication process is: the publishing event of transition $t_1$, $t_2$, $t_3$, $t_4$ are subscribed by $t_{5,1}$, the publishing event of $t_5$ is consumed by $t_{5,1}$, the publishing event of $t_3$ is consumed by $t_{5,1}$. Also, we can assume that: (1) Each production process will produce gas. (2) Each tank has a sensor to monitor the storage capacity. (3) Each production process will use the sensor to collect the acquisition

![Figure 5. Modeling concerns of the task’s fault.](image-url)
We will analyze the required backup sensors of $C_3$ based on the collected data of residue $x_1$, petroleum coking $x_2$, and petroleum coke $x_3$ in half an hour (Figure 7(a)), the $x$-coordinate is the sampling point, the $y$-coordinate is the collected data. $x_1p$, $x_2p$, $x_3p$ are the predicted value, $x_1c$, $x_2c$, $x_3c$ are the collected value, the unit is $10^{-3}$/KG$. From the actual collected data, we can get that only two collected data of sensor have failed. Which are the 16th collection point (4.8 minutes) of raw material residue and the 60th collection point (18 minutes) of cooking oil. The interval time of two failures is more than 10 minutes (13.2 minutes). When the system fails, the residue sensor will become available again, we only need introduce a backup sensor.

The purpose of Experiment 1 is to explain the effectiveness of task failure concern. Let the required reliability be $99.9\%$: (1) The system will randomly generate 480 tasks, 160 sensors and 60 connectors, then divide them into 10 groups. Each group corresponds to the element of material balancing system, and does Step 2. (2) Constructing the core net of material balance system and computing the complete task set. (3) Computing the number of backup device for each task according to the corresponding computation formula. (4) Computing the reliability of CPS according to the results of Step (3).

The result of Experiment 1 is shown in Figure 7(b), the $x$-coordinate is the different group, the $y$-coordinate is the reliability. $R$ is the required reliability, $2-B$ is the reliability by using 2 backup devices for each task.

From the figure, we can get: (1) When all tasks use 2 backup devices, the reliability of 10 groups have been improved, but it still fails to meet the required reliability. (2) The required reliability can be met by using reliability assurance strategy and the average reliability of 10 groups is $99.9752\%$. Therefore, the method can not only meet the required reliability, but also reduce the number of backup device of task.
6. Related Works

As for an emerging research area, many researchers have carried out the related works for CPS. But most of the studies are focused on the design architecture of CPS [7], many researchers have realized the importance of studying CPS [8,9]. However, it is less involved in the design and analysis process, and there is no clear consensus on the features of CPS. We will introduce some related works in this section.

The concept of event has been investigated in several contexts both within the cyber domain and the physical domain [10]. The authors introduce the concept of observers and a hierarchical spatio-temporal event model for CPS in [11]. Later, the authors introduce the concept lattice-based event model for CPS [12]. Reference [13] proposes the trustworthiness model to measure the sensor reliability and alarm confidence. However the approaches defined in the above don’t consider the specific characteristics of CPS, such as physical and computational control. Meanwhile, they ignore the analysis of feasibility and effectiveness.

The application of formal methods in the aspect orientation has got attention in the academic. Reference [14] defines model-to-model transformation rules to automatically generate either aspect-oriented or object-oriented UML 2.0 models, closing the gap between ADLs and the notations used at the detailed design phase. The authors in [15] provide a survey of existing research on aspect-oriented modeling and code generation to discover current work and identify needs for future research. A discrete computation of aspects using operational semantics by excluding the dependencies is shown in [16]. The above works use aspect-orientation and formal method to model and analyze the different areas, but these works are not related to the dynamic and static characteristics, physical and computational control of CPS, the non-functional attributions and the discrete and continuous characteristics are not considered too.

7. Conclusions

In this paper, we have proposed an aspect oriented approach to the problem of modeling and analyzing reliable CPS. In its full development, the proposed method has a rich semantics with many useful mathematical properties. The example and simulation results show that the method can achieve the following results:

1. It provides uniformity in treating different components in CPS. It can correctly characterize the static and dynamic characteristics, physical and computational control. The constructed model can effectively describe the different components of CPS.

2. It improves the flexibility of composition process, we can add or subtract the component, device based on the actual requirements.

3. It is based on a domain theoretic semantics of Petri net, which allows the representation of the essential properties and abstract many domain specifics. The operation semantics and related theories of Petri nets help prove the effectiveness and correctness of reliability assurance strategy, thereby getting the CPS with the highest reliability.

4. It has the advantages of using AOP ideas, the state space of core net and reliability model will grow with the scale of system increasing. While we only use core net in analyzing the functional properties of CPS. The reliability model is used when it comes to analyze the reliability of system.

This paper presents some preliminary results in modeling and analyzing the reliability of CPS. Several related topics are not covered in this paper, such as performance evaluation, dependency between the aspect models, as well as tool support. We will investigate these issues in the future.

Acknowledgment

The work is partially supported by the NSF of China under grants No. 61173048 and 61300041. Innovation Program of Shanghai Municipal Education Commission under Gran No. 12YZ166. The Fundamental Research Funds for the Central Universities.

References


Manuscript Received: Mar. 6, 2013
Accepted: Jan. 16, 2014