Hindawi Publishing Corporation
Discrete Dynamics in Nature and Society
Volume 2014, Article ID 825313, 14 pages
http://dx.doi.org/10.1155/2014/825313

Research Article

Confusion Analysis and Detection for Workflow Nets

Xiao-liang Chen, Zhong-yuan Jiang, and Jian-hong Ye

School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China

Correspondence should be addressed to Xiao-liang Chen; xdxlchen@gmail.com

Received 11 January 2014; Revised 29 March 2014; Accepted 30 March 2014; Published 7 May 2014

Academic Editor: Victor S. Kozyakin

Option processes often occur in a business procedure with respect to resource competition. In a business procedure modeled with a workflow net (WF-net), all decision behavior and option operations for business tasks are modeled and performed by the conflicts in corresponding WF-net. Concurrency in WF-nets is applied to keep a high-performance operation of business procedures. However, the firing of concurrent transitions in a WF-net may lead to the disappearance of conflicts in the WF-net. The phenomenon is usually called confusions that produces difficulties for the resolution of conflicts. This paper investigates confusion detection problems in WF-nets. First, confusions are formalized as a class of marked subnets with special conflicting and concurrent features. Second, a detection approach based on the characteristics of confusion subnets and the integer linear programming (ILP) is developed, which is not required to compute the reachability graph of a WF-net. Examples of the confusion detection in WF-nets are presented. Finally, the impact of confusions on the properties of WF-nets is specified.

1. Introduction

The workflow net (WF-net) [1, 2] of a business procedure is constructed to depict the logical relations among business tasks and to detect the potential faults in the procedure. Multithreaded tasks and option processes in a business procedure are usually implemented by concurrent and conflicting operations that can promote exact and efficient execution of the procedure. In its WF-net model, concurrent structures and conflicts are constructed to model and analyze multithreaded tasks and option processes, respectively.

However, WF-nets suffer from a class of faults that are called confusions such that the disappearance of conflicts occur, which implies that the conflicting transitions in a conflict become disabled before the resolution of the conflict in a WF-net. Confusions are caused by the interlacements between concurrent and conflicting processes, which generate nondeterministic occurrence of conflicts and produce difficulties for the resolution of conflicts.

The subnet shown in Figure 1(a) is a confusion that may occur in WF-nets, which includes two concurrent transitions t_1 and t_2 at the shown marking and a structural conflict between transitions t_2 and t_3. The confusion may lead to the disappearance of the conflict between t_2 and t_3, which is undesired. The defect is illustrated in the example. If the two concurrent transitions t_1 and t_2 fire, the marking shown in Figure 1(d) can be reached. In this process, different sequential transition firing sequences ($\sigma_1 = t_1t_2$ and $\sigma_2 = t_2t_1$) with respect to the concurrent firing of t_1 and t_2 implicate completely different conflict behavior. Suppose that there are two observers who observe the firing of the sequences $\sigma_1 = t_1t_2$ and $\sigma_2 = t_2t_1$, respectively. Their observation for system behavior can be described as follows.

(1) t_1 fires first, which results in a conflict between t_2 and t_3. Then, the conflict is resolved in favor of t_2 and t_2 fires ($\sigma_1 = t_1t_2$).

(2) t_2 fires first, which does not conflict with other transitions. Then, t_1 fires ($\sigma_2 = t_2t_1$).

There occurs the conflict between t_2 and t_3 and the conflict can be resolved by firing transition t_2 in observation 1. However, observation 2 shows that no conflict occurs except sequentially firing t_2 and t_1. Hence, the confusion leads to the fact that the occurrence of the conflict between t_2 and t_3 is nondeterministic owing to the concurrent firing between t_1 and t_2. The conflict behavior described in observation 1 will not occur; that is, the disappearance of the conflict occurs, when the system fires the sequence $\sigma_2 = t_2t_1$.

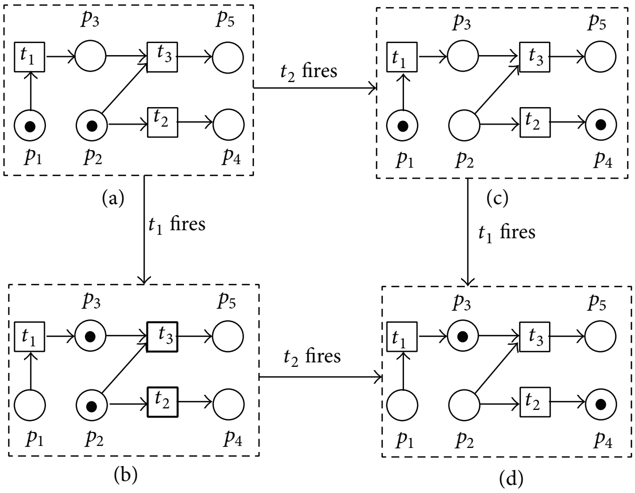


Figure 1: (a) A subnet system, (b) the subnet system obtained by firing the transition t_1 in Figure 1(a), (c) the subnet system obtained by firing the transition t_2 in Figure 1(a), and (d) the subnet system obtained by concurrently firing the transitions t_1 and t_2 in Figure 1(a).

In a real-world workflow system, the construction of conflicts and concurrency usually attempts to reach a kind of balance between the resource utilization and the fulfillment of the performance requirements. Such balance can be achieved through the timing constraints reported in [3] and the bottlenecks of performance a workflow mentioned in [4]. However, the meaningful balance can only be presented between independent conflicts and concurrency; that is, the conflicts and concurrency contain different transitions in a WF-net, since their behavior will not be mutually interfered. Conversely, if there are dependent conflicts and concurrency in a WF-net, that is, confusions, conflicts can only be considered rather than the balance between conflicts and concurrency. The reason can be presented by considering the different importance of conflicts and concurrency in a workflow procedure.

The significance of conflicts lies in the fact that they perform all option processes that are usually considered as the procedure interfaces to its external environment, for example, switches and decisions of humans. The interfaces should be determined such that the external environment can intervene a target result. Hence, the occurrence of conflicts should be observable and unambiguous in the evolution of a workflow procedure.

Concurrency in a workflow procedure only facilitates the implement of performance requirements to obtain a desired running period, which does not contribute to the logical judgments and the correctness of the procedure. Assuming that the tasks in a procedure are not performed concurrently, researchers can still get an execution result except a higher execution performance. However, if there are no conflicts, the behavior of the procedure may not be completed or the procedure performing a desired work cannot be constructed.

As aforementioned, a policy is required to determine whether a WF-net will be such an ambiguous conflict owing to concurrency. Hence, this paper aims at dealing with the formal description of confusions and providing a policy to decide whether a WF-net is confusion-free and where may appear confusions in the WF-net.

Petri nets (PNs) [5–7] as a formalism to describe the behavior of discrete event systems have drew much attention from different areas, for example, workflow management [2, 8, 9], web services [10, 11], and flexible manufacturing systems (FMSa) [12, 13]. Confusion problems and their defects were first investigated in PNs by Rozenberg et al. in [14, 15]. Their work considers the defects of PNs with confusions and mentions that it is not always possible to avoid confusion in a PN system although conflict behavior in the system with confusions is difficult to analyze. This is due to the fact that the intermediate states, for example, the markings shown in Figures 1(b) and 1(c), determined by different sequential transition sequences, for example, t_1t_2 and t_2t_1, of concurrent transitions, are different from each other.

If one requires to determine whether a conflict occurs in a PN with confusions, all possible sequential transition sequences generated by concurrent transitions in the PN have to be analyzed. However, the number of these sequences grows quickly with the increase of concurrent transitions. On the other hand, a confusion cannot be analyzed by using a reachability graph since independent concurrency (no concurrent transitions belonging to conflicting transitions) and concurrency in confusions cannot be differentiated in the reachability graph of a PN.

Some studies are devoted to the analysis of the confusions in PNs, where the phenomenon of confusions is extensively reported by many PN applications such as workflow nets (WF-nets) [2, 8, 9], occurrence nets [16, 17], safe nets [18–20], generalized (unsafe) nets [19], and generalized stochastic PNs (GSPNs) [21, 22].

The study on occurrence nets focuses on the PNs whose behavior can be interpreted by branching unfolding semantics [23]. In PNs, the behavior such as sequences, concurrency, conflicts, trails, choices, and alternatives can be described and analyzed by decomposing an occurrence net into substructures given by the node relations associated with the behavior. However, confusions cannot be described by the existing branching semantics. Hence, Smith and Haar consider the independence of events in occurrence nets and the indirect influences among concurrent events in [16, 17]. Furthermore, interference structural conflict clusters are developed in [17] in order to describe confusions. The clusters belong to a kind of the substructures of occurrence nets. In the work of Smith and Haar, confusion detection problems are not considered.

In [18, 19], a confusion detection approach based on trace theory is proposed. Furthermore, communicating sequential process (CSP) model checker is used to verify the detected confusions. However, the precondition of the confusion detection is that a PN is converted into an interleaving process algebra CSP. The transformation can produce additional costs and is infeasible for an arbitrary PN. A feasible offline confusion detection policy is reported in [20], which requires computing the reachability graph of a PN and cannot be used in complex systems owing to state space explosions.

Confusions in GSPNs are discussed in [21, 22]. The marking graph of a GSPN is not a stochastic process if the GSPN contains confusions, which implies that continuous-time Markov chains (CTMCs) cannot be used to analyze the GSPNs with confusions. Generally, a classical analytical approach for GSPNs is to assume that the subnets of immediate transitions are confusion-free in order that the analysis can proceed. However, the assumption does not intrinsically solve the problem of confusions since they really exist in GSPNs.

A PN which models the control-flow dimension of a workflow is a WF-net. Both dynamic behavior and control of a case in a workflow can be specified in its WF-net [1, 24]. van der Aalst and Hee deal with the existence of confusions in WF-nets and show the defects of WF-nets with confusions [2, 8, 9]. First, a WF-net is said to be correct with a singleton token in its source place if the WF-net holds for the properties "soundness" and "well-structured," which is reported in [9]. If a WF-net is sound, an arbitrary reachable marking generated by the WF-net can terminate properly. A well-structured WF-net has a number of nice properties. For example, the soundness of a WF-net can be verified in polynomial time and a sound well-structured WF-net is safe. However, the existence of confusions in WF-nets leads to nondeterministic conflicting processes. Hence, it is not amenable to say that a sound or a well-structured WF-net is correct since the properties "soundness" and "well-structured" in WF-nets cannot avoid confusions. Furthermore, confusions can affect the property "well-structured" of a WF-net under the acyclic (no directed cycle in the structure of a WF-net) restrictions. To be more specific,

(1) a sound and well-structured WF-net with cycles may cause confusions; this fact will be illustrated by giving an example in this paper;

(2) if an acyclic sound WF-net contains a confusion, it is certainly not well structured; the conclusion will be specified in this work.

Second, in a workflow management system (WFMS), four important building blocks (AND-split, AND-join, OR-split, and OR-join) are introduced to specify the functions of the WFMS, where OR-splits are said to be case variables and modeled with conflicts in the corresponding WF-system, which lead to different routings of a case (transition sequences). According to the requirements of WFMSs and the work of van der Aalst in [9], the routings of a case should be independent of the order in which tasks are executed; that is, the choice between conflicting transitions cannot be influenced by the order in which a nonconflicting transition fires. It is shown that the behavior of confusions should not occur in a WF-system modeling a WFMS.

Third, in WF-net, only free choice is usually allowed due to the existence of confusions since the structures of confusions are nonfree choice. However, as discussed in [2], there are nonfree choice WF-nets that correspond to sensible workflows. They can be applied if confusion detection problems are solved.

Fault detection and control in PNs is an important and computationally difficult problem owing to resource sharing, state space explosions, and complex conflicting and concurrent behavior. For example, some excellent deadlock detection and prevention methods of FMSs are proposed in PNs [25–31]. WF-nets belong to a subclass of PNs and the occurrence of confusions in a WF-net is undesired. Hence, this paper is intended to investigate the confusion problems in a WF-net. The main contributions include the following three aspects.

(1) The formalization of confusions is presented by defining a special class of subnets.

(2) The impact of confusions on the properties of WF-nets is discussed.

(3) A confusion detection policy is proposed without considering the reachability graph of WF-nets, which can return whether and where may occur a confusion in WF-nets.

If a sound or well-structured WF-net is reported to be confusion-free according to the proposed detection policy, the WF-net can be used in any real-world workflow procedure. Correspondingly, once the proposed policy finds a confusion in WF-nets, researchers or engineers will be able to give up the use of such WF-nets or design a method to ensure the occurrence of conflicts before using them.

This paper is organized as follows. Section 2 introduces basic concepts of PNs and WF-nets. Section 3 formulates two classes of confusions and specifies their behavior by some examples. Section 4 presents algorithms for confusion detection. The proposed algorithms are applied to some examples and a real-world case. Section 5 deals with the impact of confusions on the properties of WF-nets. Conclusions and future work are presented in Section 6.

2. Basics of Petri Nets and Workflow Nets

This section provides the basics of PNs and WF-nets, where the definitions and the properties of PNs presented in this paper come from [6] and the details of WF-nets can be found in [9].

A PN N is a four-tuple (P, T, F, W), where P and T are finite, nonempty, and disjoint sets. P is the set of places and T is the set of transitions. $F \subseteq (P \times T) \cup (T \times P)$ is called a flow relation of the net, represented by arcs with arrows from places to transitions or from transitions to places. $W : (P \times T) \cup (T \times P) \rightarrow \mathbb{N}$ is a mapping that assigns a weight to an arc: $W(x, y) > 0$ if $(x, y) \in F$, and $W(x, y) = 0$ otherwise, where $x, y \in P \cup T$ and $\mathbb{N} = \{0, 1, 2, \ldots\}$ is a set of nonnegative integers. A net is self-loop-free (pure) if $\nexists x, y \in P \cup T, (x, y) \in F \wedge (y, x) \in F$. $N = (P, T, F, W)$ is called an ordinary net, denoted as $N = (P, T, F)$, if $\forall f \in F, W(f) = 1$.

Let $x \in P \cup T$ be a node in a net $N = (P, T, F, W)$. The preset of x is defined as ${}^{\bullet}x = \{y \in P \cup T \mid (y, x) \in F\}$, while the postset of x is defined as $x^{\bullet} = \{y \in P \cup T \mid (x, y) \in F\}$. Let X be a set of nodes with $X \subseteq P \cup T$. We have ${}^{\bullet}X = \cup_{x \in X} {}^{\bullet}x$, and $X^{\bullet} = \cup_{x \in X} x^{\bullet}$. The structure of a self-loop-free PN can be represented by its incidence matrix $[N]$; that is, a $|P| \times |T|$ integer matrix with $[N](p, t) = W(t, p) - W(p, t)$.

A marking M of N is a mapping from P to $\mathbb{N}$. $M(p)$ denotes the number of tokens contained in place p. p is marked at marking M if $M(p) > 0$. (N, M_0) is called a net system or a marked net and M_0 is called an initial marking of N. A marking M can be denoted by $\sum_{p\in P} M(p)p$ that is presented as multiset or formal sum notations. For example, a marking $M = (2, 0, 2, 4)^T$ in a net with four places can be denoted by $M = 2p_1 + 2p_3 + 4p_4$.

A transition $t \in T$ is enabled at marking M if $\forall p \in {}^\bullet t$, $M(p) \geq W(p,t)$, which is denoted as $M[t\rangle$. If t is enabled, it can fire. Its firing yields another marking M' such that $\forall p \in P, M'(p) = M(p) - W(p,t) + W(t,p)$, which is denoted by $M[t\rangle M'$. Marking M' is said to be reachable from M if there exists a transition sequence $\sigma = t_1t_2\cdots t_n$ and markings $M_1, M_2, \ldots$, and M_{n-1} such that $M[t_1\rangle M_1[t_2\rangle M_2 \cdots M_{n-1}[t_n\rangle M'$. This is denoted by $M[\sigma\rangle M'$. The set of markings reachable from M in N defines the reachability set of (N, M), denoted as $R(N, M)$.

A transition $t \in T$ in a PN system (N, M_0) is live at M_0 if $\forall M \in R(N, M_0), \exists M' \in R(N, M_0), M'[t\rangle$ holds. The system (N, M_0) is live if $\forall t \in T$, t is live at M_0. A PN (N, M_0) is safe if $\forall M \in R(N, M_0), \forall p \in P, M(p) \leq 1$. Generally, safe PNs are used to depict and analyze business processes, logistic systems, and workflow management systems. In a safe PN, a transition $t \in T$ is enabled at marking M if $\forall p\in {}^\bullet t$, $M(p) = 1$, and $\forall p \in t^\bullet$, $M(p) = 0$.

A P-vector is a column vector $I : P \rightarrow \mathbb{Z}$ indexed by P, where $\mathbb{Z}$ is the set of integers. I is a P-invariant if $I \neq \mathbf{0}$ and $[N]^T \cdot I = \mathbf{0}$, where the bold $\mathbf{0}$ denotes a column vector whose every entry equals 0. Let I be a P-invariant of N. Then $\forall M \in (R, M_0)$; $I^T \cdot M = I^T \cdot M_0$ holds. $\|I\| = \{p \mid I(p) \neq 0\}$ is called the support of I. I is called a minimal P-invariant if $\|I\|$ is not a superset of the support of any other one and its components are mutually prime.

Let $N = (P, T, F)$ be a PN and $n_1, n_k \in P \cup T$ be two nodes in N, where $k \in \{1, 2, \ldots, |P \cup T|\}$. A string $C = n_1n_2\cdots n_k$ is called a path of N if $\forall i \in \{1, 2, \ldots, k-1\}$, $(n_i, n_{i+1}) \in F$. $C = n_1n_2\cdots n_k$ is called a directed path if $\forall i \in \{1, 2, \ldots, k-1\}$, $n_{i+1} \in n_i^\bullet$. $C = n_1n_2\cdots n_k$ is called elementary path if for any two nodes n_i and n_j in C $i \neq j \Rightarrow n_i \neq n_j$.

Definition 1 (see [9]). A PN $N = (P, T, F)$ is said to be a workflow net (WF-net) if and only if

(1) N has a source place $p_{\text{in}} \in P$ and a sink place $p_{\text{out}} \in P$ such that ${}^\bullet p_{\text{in}} = \emptyset$ and $p_{\text{out}}^\bullet = \emptyset$.

(2) For any node $x \in P \cup T$, x belongs to a directed path from p_{in} to p_{out}.

(N, M_{in}) is said to be a workflow system (WF-system) if M_{in} is the initial marking that contains only a singleton token in place p_{in} in the system.

In this paper, "WF-net" and "WF-system" are used to distinguish an arbitrary net structure N and the structure N at initial marking M_{in}, that is, (N, M_{in}), respectively. WF-nets are a subclass of PNs. Hence, any analytical methods involved in PNs can be applied to WF-nets. In this paper, safe WF-system is considered only.

Table 1: Description of a business procedure in a WF-system.

Business process	WF-system
Task	Transition
Condition	Place
Start condition	Marked source place p_{in}
End condition	Marked sink place p_{out}
Live task	Live transition in extended WF-system
Dead task	Dead transition in extended WF-system
Case	Path from p_{in} to p_{out}

Definition 2 (see [9]). A PN $\widetilde{N} = (\widetilde{P}, \widetilde{T}, \widetilde{F})$ is said to be the extended WF-net from a WF-net $N = (P, T, F)$ if an extra transition t^* exists such that $\widetilde{P} = P$, $\widetilde{T} = T \cup \{t^*\}$, and $\widetilde{F} = F \cup \{(p_{\text{out}}, t^*), (t^*, p_{\text{in}})\}$. $(\widetilde{N}, M_{\text{in}})$ is called an extended WF-system if (N, M_{in}) is a WF-system and $\widetilde{N}$ is extended from N.

The description of a business procedure in a WF-system is shown in Table 1.

The life cycle of a business procedure is depicted by the logical relations among transitions in a WF-net. Marked source place p_{in} and sink place p_{out} in a WF-system are used to identify the beginning and the termination of a business procedure. A task in a business procedure contains three status: *idle*, *ready*, and *finished*, which implies that a transition in corresponding WF-system is disabled, enabled, and fires, respectively.

A WF-system cannot allow the retention of tokens in other places if p_{out} is marked, which is ensured by condition (2) in Definition 1. Figures 2(a) and 2(b) show a WF-system (N, M_{in}) at initial marking $M_{\text{in}} = (1, 0, 0, 0, 0, 0)^T$ and its extended WF-system $(\widetilde{N}, M_{\text{in}})$, respectively.

Definition 3 (see [9]). A WF-system (N, M_{in}) with $P = \{p_{\text{in}}, p_1, p_2, \ldots, p_n, p_{\text{out}}\}$ is said to be sound if

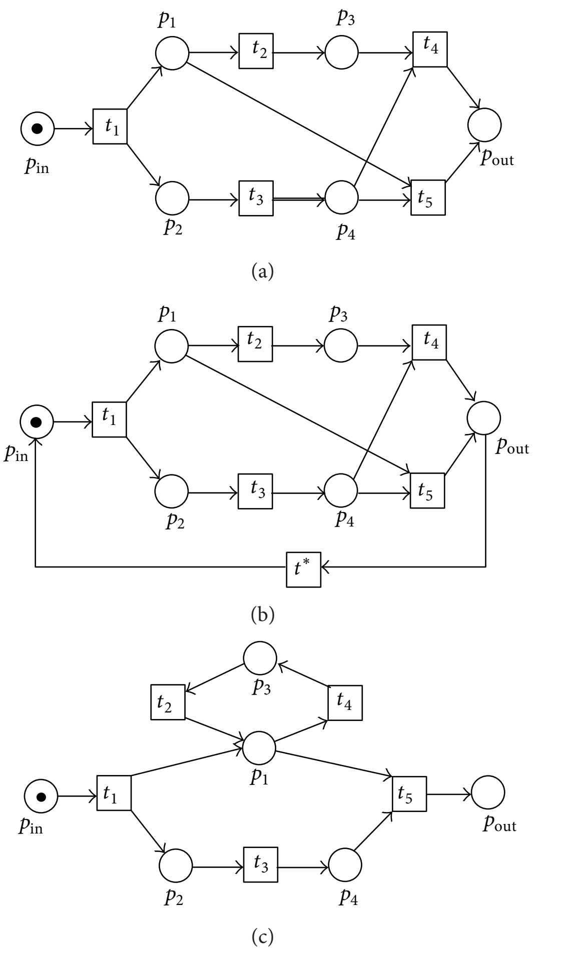
(1) $\forall M \in R(N, M_{\text{in}})$, $M_{\text{in}}[\sigma_1\rangle M \Rightarrow \exists \sigma_2$, $[\sigma_2\rangle M_{\text{out}}$, where $M_{\text{in}} = (1, \underbrace{0, 0, \ldots, 0}_{n+1})^T$ and $M_{\text{out}} = (\underbrace{0, 0, \ldots, 0}_{n+1}, 1)^T$.

(2) $\forall M \in R(N, M_{\text{in}})$, $M_{\text{in}}[\sigma_1\rangle M \wedge M \geq M_{\text{out}} \Rightarrow M = M_{\text{out}}$.

(3) $\forall t \in \widetilde{T}, \forall M \in R(\widetilde{N}, M_{\text{in}}), \exists M' \in R(\widetilde{N}, M_{\text{in}}), M'[t\rangle$ holds.

Condition (1) in Definition 3 guarantees that the marking M_{out} in a sound WF-system can be reached by firing an existing transition sequence at an arbitrary reachable marking, which implies that any case in a business procedure can be terminated eventually. Condition (2) in Definition 3 ensures that marking M_{out} is the only terminal marking. The fact that no dead transitions exist in the extended WF-system $(\widetilde{N}, M_{\text{in}})$ of (N, M_{in}) is ensured by condition (3) in Definition 3.

A WF-system (N, M_{in}) is sound if and only if the extended WF-system $(\widetilde{N}, M_{\text{in}})$ of (N, M_{in}) is live and



(a)

(b)

(c)

Figure 2: (a) A sound WF-system (N, M_{in}), (b) the extended WF-system $(\widetilde{N}, M_{\text{in}})$ of (N, M_{in}), and (c) a sound and well-structured WF-system (N', M'_{in}).

bounded. For example, the WF-system shown in Figure 2(a) is sound since its extended WF-system shown in Figure 2(b) is live and bounded.

Definition 4 (see [9]). A PN N is said to be well handled if for any pair of nodes $x \in P \cup T$ and $y \in P \cup T$ such that one of the nodes is a place and the other is a transition and for any pair of elementary paths C_1 and C_2 from x to y; the formula $\alpha(C_1) \cap \alpha(C_2) = \{x, y\} \Rightarrow C_1 = C_2$ holds, where $\alpha(C_1)$ and $\alpha(C_2)$ denote the set of all nodes in path C_1 and C_2, respectively. A WF-net N is well structured if the extended $\widetilde{N}$ from N is well handled.

The WF-system (N, M_{in}) shown in Figure 2(a) is not well structured and the WF-system (N', M'_{in}) shown in Figure 2(c) is well structured. A well-structured WF-net has a number of nice properties. For example, the soundness of a WF-system can be verified in polynomial time and a sound well-structured WF-system is safe. A reasonable WF-system usually holds for soundness and well structure.

3. Conflicts, Concurrency, and Confusions

This section formalizes confusions in a PN system to a class of marked subnets with respect to special conflicting and concurrent characteristics. The formalization allows us to obtain a confusion detection policy in WF-systems.

Definition 5 (see [32]). A structural conflict in a net $N = (P, T, F)$ is a pair $K = \langle p, T(K)\rangle$, where $p \in P$ is called the structural conflicting place and $T(K) = \{t_1, t_2, \ldots, t_n\}$ is the set of output transitions of p with $n = |p^\bullet| \geq 2$. The elements in $T(K)$ are called structural conflicting transitions.

Definition 6 (see [32]). An effective conflict in a marked net (N, M_0), denoted by $K^M = \langle p, T(K^M), M\rangle$, is associated with a structural conflict $K = \langle p, T(K)\rangle$ and a marking $M \in R(N, M_0)$ such that $\forall t \in T(K^M)$, $M[t\rangle$ is true, and $M(p) < \sum_{t\in T(K^M)} W(p,t)$, where $T(K^M) \subseteq T(K)$ is the set of enabled transitions in $T(K)$ at marking M.

The cardinality of $T(K^M)$, that is, $|T(K^M)|$, is said to be the size of effective conflict K^M. An effective conflict is said to be complete (resp., incomplete) if $|T(K^M)| = |T(K)|$ (resp., $1 \leq |T(K^M)| < |T(K)|$). Figures 3(a) and 3(b) show a structural conflict $K = \langle p_2, \{t_1, t_2, t_3, t_4\}\rangle$ and an effective conflict $K^M = \langle p_2, \{t_2, t_3, t_4\}, p_2 + p_3\rangle$ associated with K, respectively, where $T(K) = \{t_1 - t_4\}$ and $T(K^M) = \{t_2 - t_4\}$. The effective conflict K^M is incomplete since $|T(K^M)| = 3 < |T(K)| = 4$.

Definition 7. (1) Transitions t_1 and t_2 are said to be concurrent at marking M in a marked net (N, M) if $M[t_1\rangle$, $M[t_2\rangle$, and $({}^\bullet t_1 \cup t_1^\bullet) \cap ({}^\bullet t_2 \cup t_2^\bullet) = \emptyset$.

(2) Let $D^M = \{t_1, t_2, \ldots, t_n\}$ be a transition set. It is said to be a set of concurrent transitions at marking M if $\forall t_i, t_k \in D^M$, $i \neq k$; transitions t_i and t_k are concurrent at marking M.

(3) The concurrent degree of a set D^M of concurrent transitions, denoted by λ, is defined as $\lambda = |D^M|$.

(4) Let D^M be a set of concurrent transitions in a marked net (N, M) with $\lambda = |D^M|$. It is said to be λ-max-concurrent in (N, M) if there does not exist a set of concurrent transitions $\widehat{D}^M \neq D^M$ with $\widehat{\lambda} = |\widehat{D}^M|$ in (N, M) such that $\widehat{\lambda} > \lambda$. A transition in D^M is said to be λ-max-concurrent if D^M is λ-max-concurrent in (N, M).

(5) Let $\mathbb{D}^M = \{D_1^M, D_2^M, \ldots, D_\rho^M\}$ be the set of concurrent transition sets in (N, M); that is, $\forall D_j^M \in \mathbb{D}^M$; D_j^M is a set of concurrent transitions in (N, M). $\mathbb{D}^M$ is said to be λ-max-concurrent if $\forall D_j^M \in \mathbb{D}^M$; D_j^M is λ-max-concurrent in (N, M).

Figure 3(c) shows a marked PN that contains totally seven sets $D_1^M = \{t_1, t_4\}$, $D_2^M = \{t_1, t_5\}$, $D_3^M = \{t_1, t_6\}$, $D_4^M = \{t_4, t_5\}$, $D_5^M = \{t_4, t_6\}$, $D_6^M = \{t_1, t_4, t_5\}$, and $D_7^M = \{t_1, t_4, t_6\}$ of concurrent transitions at marking $M = p_1 + p_4 + p_5$. Two sets D_6^M and D_7^M with $\lambda_6 = \lambda_7 = 3$ are 3-max-concurrent since there does not exists a set of concurrent transitions from D_1^M to D_7^M, whose cardinality is greater than three. Suppose that $\mathbb{D}_1^M = \{D_6^M, D_7^M\}$ and $\mathbb{D}_2^M = \{D_1^M, D_2^M, D_7^M\}$

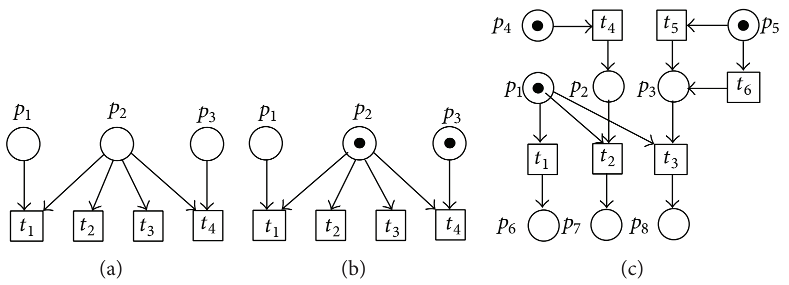


FIGURE 3: (a) A structural conflict $K = \langle p_2, \{t_1 - t_4\}\rangle$, (b) an effective conflict $K^M = \langle p_2, \{t_2 - t_4\}, p_2 + p_3\rangle$, and (c) illustration of concurrency.

are two sets of concurrent transition sets. Then, $\mathbb{D}_1^M$ is 3-max-concurrent and $\mathbb{D}_2^M$ is not 3-max-concurrent since the concurrent degrees of D_1^M and D_2^M in $\mathbb{D}_2^M$ are less than three.

Definition 8. A marked PN $(\overline{N} = (\overline{P}, \overline{T}, \overline{F}), \overline{M})$ is said to be a marked subnet of a PN system $(N = (P, T, F), M)$ if $\overline{P} \subseteq P$, $\overline{T} \subseteq T$, $\overline{F} = F \cap [(\overline{P} \times \overline{T}) \cup (\overline{T} \times \overline{P})]$, and $\overline{M}$ is a natural projection of marking M on place set $\overline{P}$. $\overline{M}$ is said to be valid (resp., invalid), denoted by $\overline{M} \prec M$ (resp., $\overline{M} \nprec M$), if $\exists p \in \overline{P}$, $\overline{M}(p) = 1$ (resp., $\forall p \in \overline{P}$, $\overline{M}(p) = 0$).

Definition 9. A confusion, denoted by $(\overline{N}, T(K), \overline{M}, \mathbb{D}^{\overline{M}})$ or abbreviated to $(\overline{N}, \overline{M})$, is a marked subnet in a net system (N, M_0) such that

(1) there exists a structural conflict $K = \langle p, T(K)\rangle$ with $p \in \overline{P}$ and $T(K) \subset \overline{T}$;

(2) there exists a nonempty set $\mathbb{D}^{\overline{M}}$ that is composed of all 2-max-concurrent transition sets at the submarking $\overline{M}$ in $(\overline{N}, \overline{M})$;

(3) $\forall D_j^{\overline{M}} \in \mathbb{D}^{\overline{M}}$, $\exists t_i \in D_j^{\overline{M}} \setminus T(K)$ such that $\overline{M}[t_i\rangle\overline{M}'$ and $|T(K^{\overline{M}})| \neq |T(K^{\overline{M}'})|$;

(4) there exist two reachable markings M and M' in $R(N, M_0)$ such that $\overline{M} \prec M$ and $\overline{M}' \prec M'$ are true.

A confusion is said to be a conflict-increasing confusion (CIC) if $|T(K^{\overline{M}})| < |T(K^{\overline{M}'})|$. It is said to be a conflict-decreasing confusion (CDC) if $|T(K^{\overline{M}})| > |T(K^{\overline{M}'})|$.

Confusions are tackled as the marked subnets in a PN system, which are distinguished into CICs and CDCs. Two requirements on confusions are imposed by Conditions (1) and (2) in Definition 9, respectively. If a marked subnet $(\overline{N}, \overline{M})$ is a confusion, it necessarily contains a structural conflict and all the sets of concurrent transitions in $(\overline{N}, \overline{M})$ are 2-max-concurrent at marking $\overline{M}$.

Condition (3) in Definition 9 implies the behavior characteristics of confusions. For a structural conflict K in a confusion, there exists at least a transition t not belonging to $T(K)$ in the confusion. Firing t changes the size of effective conflicts associated with K. Condition (4) in Definition 9 ensures that the behavior described in Condition (3) can actually occur in the original net system. Finally, confusions are classified into CICs and CDCs according to the relations of the size change of effective conflicts in Definition 9.

If a marked subnet of a PN system satisfies Definition 9, the disappearance of complete effective conflicts will occur in the marked subnet. The limitation of the 2-max-concurrency can limit the size of a subnet at a marking such that the marked subnet is exactly a CIC or a CDC. In other words, the behavior of a confusion defined in Definition 9 with the limitation of 2-max-concurrency can only be one of the two results, that is, increasing or decreasing the size of effective conflicts. Examples 1 and 2 are given to show the structures and behavior of CICs and CDCs defined in Definition 9, respectively.

Example 1. The marked subnet shown in Figure 4(a) is a CIC $(\overline{N}_1, T(K_1), \overline{M}_1, \mathbb{D}^{\overline{M}_1})$, where $T(K_1) = \{t_1, t_2, t_3, t_4\}$, $\overline{M}_1 = p_1 + p_4$, $\mathbb{D}^{\overline{M}_1} = \{D_1^{\overline{M}_1}, D_2^{\overline{M}_1}\}$, $D_1^{\overline{M}_1} = \{t_1, t_5\}$, and $D_2^{\overline{M}_1} = \{t_2, t_5\}$. The confusion holds for Conditions (1) and (2) in Definition 9. $\overline{N}_1$ contains a structural conflict $K_1 = \langle p_1, \{t_1 - t_4\}\rangle$ and two 2-max-concurrent transition sets $D_1^{\overline{M}_1}$ and $D_2^{\overline{M}_1}$ exist in the confusion such that $\mathbb{D}^{\overline{M}} \neq \emptyset$ is true.

The partial reachability graph of the confusion is shown in Figures 4(b) and 4(c), which are obtained by concurrently firing transitions in $D_1^{\overline{M}_1} = \{t_1, t_5\}$ and $D_2^{\overline{M}_1} = \{t_2, t_5\}$, respectively, where two transition firing sequences $\sigma_1 = t_1 t_5$ and $\sigma_2 = t_5 t_1$ (resp., $\sigma_3 = t_2 t_5$ and $\sigma_4 = t_5 t_2$) can be observed from the initial marking $\overline{M}_1 = p_1 + p_4$ to the terminal marking $\overline{M}_4 = p_2 + p_3 + p_5$ (resp., $\overline{M}_6 = p_2 + p_3 + p_6$) in Figure 4(b) (resp., Figure 4(c)) and any marking $\overline{M}_j$ in Figure 4 is represented with the size of the effective conflict $K_1^{\overline{M}_j}$; that is, $|T(K_1^{\overline{M}_j})|$.

According to Condition (3) in Definition 9, $(\overline{N}_1, \overline{M}_1)$ is a CIC since firing the nonconflicting transition $t_5 \notin K_1$ in $D_1^{\overline{M}_1}$ (resp., $D_2^{\overline{M}_1}$) can expand the size of the effective conflicts associated with K_1 from $|T(K_1^{\overline{M}_1})| = 2$ to $|T(K_1^{\overline{M}_3})| = |T(K_1)| = 4$, where $K_1^{\overline{M}_1} = \langle p_1, \{t_1, t_2\}, p_1 + p_4\rangle$ is incomplete and $K_1^{\overline{M}_3} = \langle p_1, \{t_1 - t_4\}, p_1 + p_2 + p_3\rangle$ is complete.

The fact whether the complete effective conflict $K_1^{\overline{M}_3}$ occurs according to the concurrent firing of transitions in

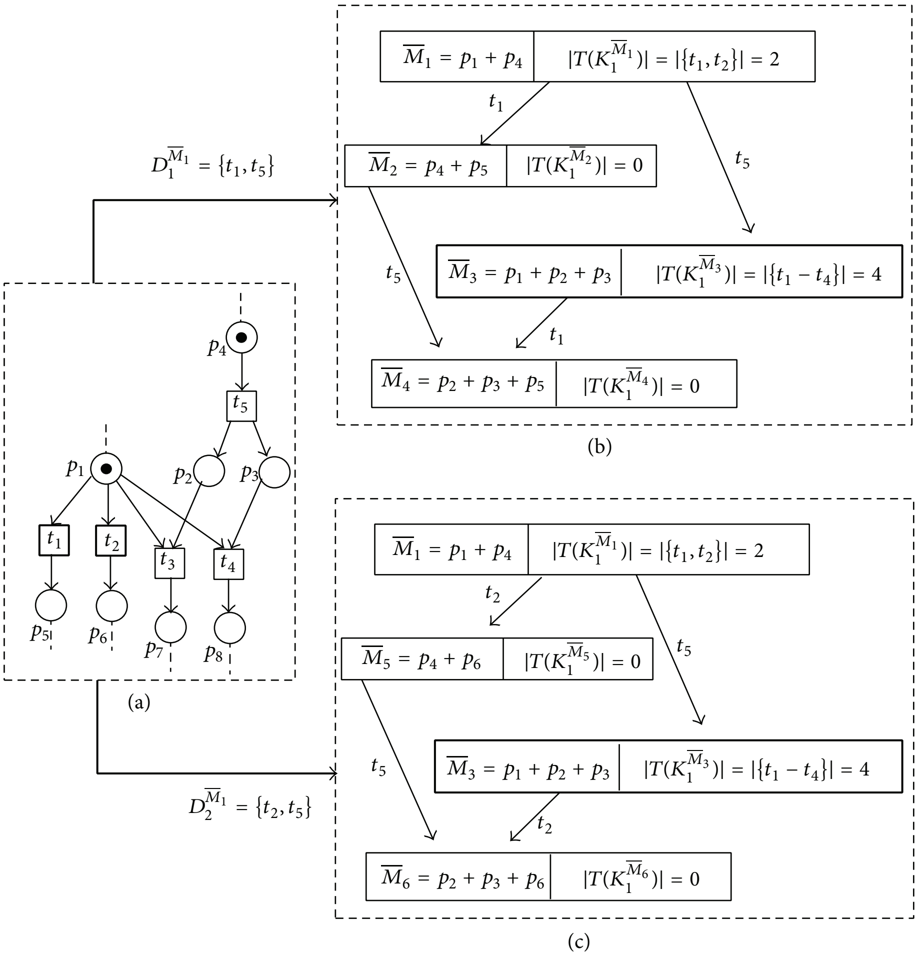

$\overline{M}_1 = p_1 + p_4$
$|T(K_1^{\overline{M}_1})| = |\{t_1, t_2\}| = 2$
t_1
t_5
$D_1^{\overline{M}_1} = \{t_1, t_5\}$
$\overline{M}_2 = p_4 + p_5$
$|T(K_1^{\overline{M}_2})| = 0$
t_5
$\overline{M}_3 = p_1 + p_2 + p_3$
$|T(K_1^{\overline{M}_3})| = |\{t_1 - t_4\}| = 4$
t_1
$\overline{M}_4 = p_2 + p_3 + p_5$
$|T(K_1^{\overline{M}_4})| = 0$
(b)
p_4
t_5
p_1
p_2
p_3
t_1
t_2
t_3
t_4
p_5
p_6
p_7
p_8
(a)
$\overline{M}_1 = p_1 + p_4$
$|T(K_1^{\overline{M}_1})| = |\{t_1, t_2\}| = 2$
t_2
t_5
$\overline{M}_5 = p_4 + p_6$
$|T(K_1^{\overline{M}_5})| = 0$
t_5
$\overline{M}_3 = p_1 + p_2 + p_3$
$|T(K_1^{\overline{M}_3})| = |\{t_1 - t_4\}| = 4$
$D_2^{\overline{M}_1} = \{t_2, t_5\}$
t_2
$\overline{M}_6 = p_2 + p_3 + p_6$
$|T(K_1^{\overline{M}_6})| = 0$
(c)

FIGURE 4: The partial reachability graph of a CIC $(\overline{N}_1, \overline{M}_1)$ with $|T(K_1)| = 4$ by concurrently firing transitions in $D_1^{\overline{M}_1} = \{t_1, t_5\}$ and $D_2^{\overline{M}_1} = \{t_2, t_5\}$.

$D_1^{\overline{M}_1} = \{t_1, t_5\}$ and $D_2^{\overline{M}_1} = \{t_2, t_5\}$ is nondeterministic. If $\sigma_2 = t_5t_1$ and $\sigma_4 = t_5t_2$ fire, the complete effective conflict $K_1^{\overline{M}_3}$ with size $|T(K_1^{\overline{M}_3})| = |T(K_1)| = 4$ at marking $\overline{M}_3 = p_1+p_2+p_3$ can occur. If $\sigma_1 = t_1t_5$ and $\sigma_3 = t_2t_5$ fire, the processes $\overline{M}_1[t_1\rangle\overline{M}_2[t_5\rangle\overline{M}_4$ and $\overline{\mathrm{M}}_1[t_2\rangle\overline{M}_5[t_5\rangle\overline{M}_6$ do not expand the size of the effective conflicts associated with K_1 such that a complete effective conflict occurs, where $\overline{M}_1 = p_1 + p_2$, $\overline{M}_2 = p_4 + p_5$, $\overline{M}_4 = p_2 + p_3 + p_5$, $\overline{M}_5 = p_4 + p_6$, and $\overline{M}_6 = p_2 + p_3 + p_6$.

The complete $K_1^{\overline{M}_3}$ is expected to occur since all the structural conflicting transitions in K_1 are enabled such that the options can be done among transitions $t_1 - t_4$. In this case, we can say that the disappearance of the effective conflict $K_1^{\overline{M}_3}$ occurs if the CIC fires the transition sequences $\sigma_1 = t_1t_5$ and $\sigma_3 = t_2t_5$.

Example 2. The marked subnet shown in Figure 5(a) is a CDC that consists of two structural conflicts $K_2 = \langle p_1, \{t_1 - t_4\}\rangle$ and $K_3 = \langle p_2, \{t_3 - t_6\}\rangle$ at the initial marking $\overline{M}_1 = p_1 + p_2$. Only K_2 in the confusion is considered due to the symmetry. In other words, the confusion $(\overline{N}_2, T(K_2), \overline{M}_1, \mathbb{D}^{\overline{M}_1})$ is analyzed, where $T(K_2) = \{t_1 - t_4\}$, $\mathbb{D}^{\overline{M}_1} = \{D_1^{\overline{M}_1}, D_2^{\overline{M}_1}, D_3^{\overline{M}_1}, D_4^{\overline{M}_1}\}$, $D_1^{\overline{M}_1} = \{t_1, t_5\}$, $D_2^{\overline{M}_1} = \{t_1, t_6\}$, $D_3^{\overline{M}_1} = \{t_2, t_5\}$, and $D_4^{\overline{M}_1} = \{t_2, t_6\}$.

The confusion is a CDC since firing transition $t_5 \notin T(K_2)$ in $D_1^{\overline{M}_1}$ and $D_3^{\overline{M}_1}$ or firing another transition $t_6 \notin T(K_2)$ in $D_2^{\overline{M}_1}$ and $D_4^{\overline{M}_1}$ reduces the size of the effective conflict $K_2^{\overline{M}_1} = \langle p_1, \{t_1, t_2, t_3, t_4\}, \overline{M}_1\rangle$ such that the options cannot be done in the complete effective conflict $K_2^{\overline{M}_1}$.

The partial reachability graph obtained by concurrently firing transitions t_1 and t_5 in $D_1^{\overline{M}_1}$ is shown in Figure 5(b). Two transition firing sequences $\sigma_1 = t_1t_5$ and $\sigma_2 = t_5t_1$ can be obtained. The complete effective conflict $K_2^{\overline{M}_1} = \langle p_1, \{t_1 - t_4\}, \overline{M}_1\rangle$ with $|T(K_2^{\overline{M}_1})| = |T(K_2)| = 4$ can be found at the initial marking $\overline{M}_1 = p_1 + p_2$. Hence, the options on conflicting transitions from t_1 to t_4 should be done for the effective conflict $K_2^{\overline{M}_1}$. However, an unfortunate transition $t_5 \notin T(K_2)$ can concurrently fire such that the size is reduced from $|T(K_2^{\overline{M}_1})| = 4$ to $|T(K_2^{\overline{M}_3})| = 2$, which leads to disabled conflicting transitions t_3 and t_4 before the options among

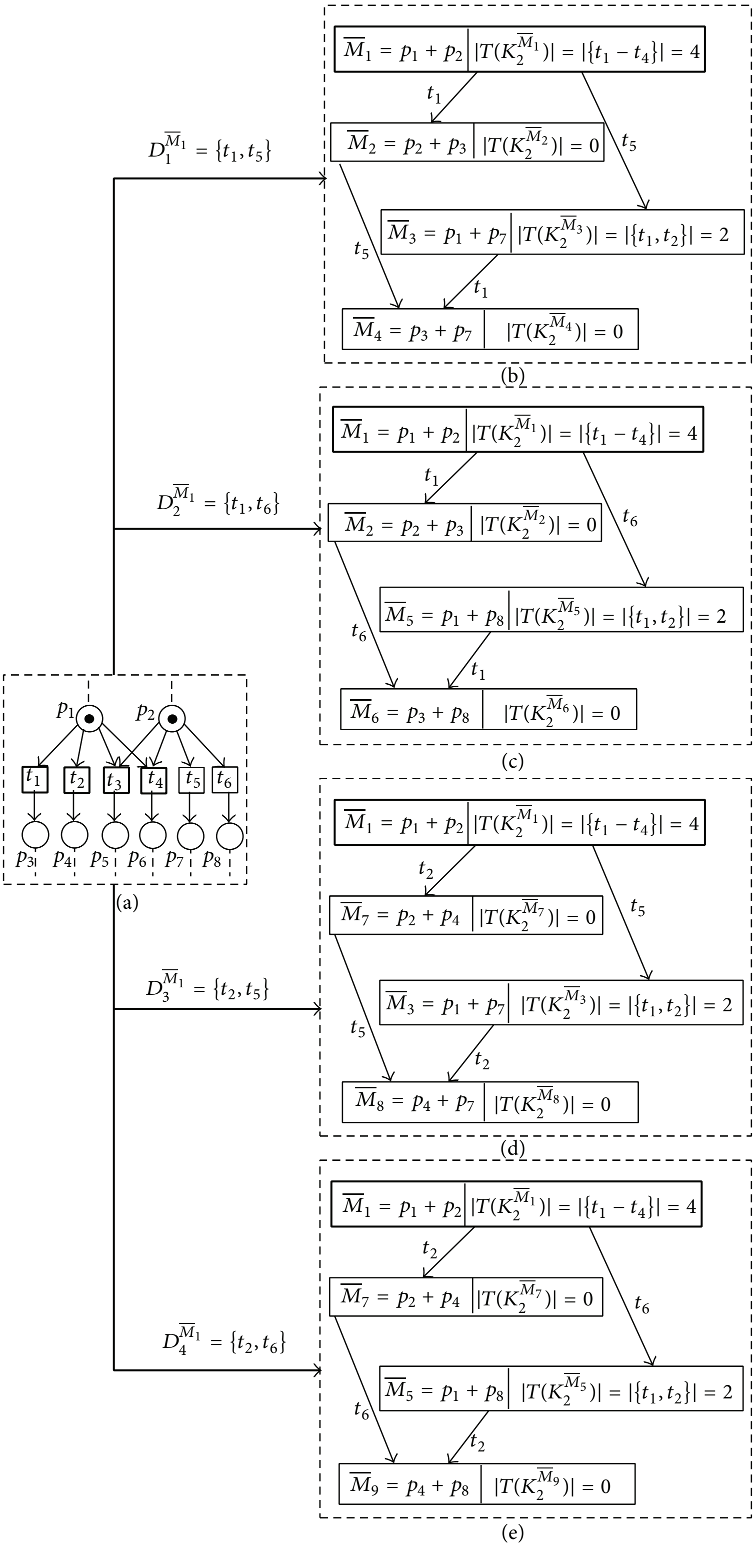

$D_1^{\overline{M}_1} = \{t_1, t_5\}$
$D_2^{\overline{M}_1} = \{t_1, t_6\}$
$D_3^{\overline{M}_1} = \{t_2, t_5\}$
$D_4^{\overline{M}_1} = \{t_2, t_6\}$
p_1
p_2
t_1 t_2 t_3 t_4 t_5 t_6
p_3 p_4 p_5 p_6 p_7 p_8
(a)
$\overline{M}_1 = p_1 + p_2$ | $|T(K_2^{\overline{M}_1})| = |\{t_1 - t_4\}| = 4$
t_1
t_5
$\overline{M}_2 = p_2 + p_3$ | $|T(K_2^{\overline{M}_2})| = 0$
$\overline{M}_3 = p_1 + p_7$ | $|T(K_2^{\overline{M}_3})| = |\{t_1, t_2\}| = 2$
t_5
t_1
$\overline{M}_4 = p_3 + p_7$ | $|T(K_2^{\overline{M}_4})| = 0$
(b)
$\overline{M}_1 = p_1 + p_2$ | $|T(K_2^{\overline{M}_1})| = |\{t_1 - t_4\}| = 4$
t_1
t_6
$\overline{M}_2 = p_2 + p_3$ | $|T(K_2^{\overline{M}_2})| = 0$
$\overline{M}_5 = p_1 + p_8$ | $|T(K_2^{\overline{M}_5})| = |\{t_1, t_2\}| = 2$
t_6
t_1
$\overline{M}_6 = p_3 + p_8$ | $|T(K_2^{\overline{M}_6})| = 0$
(c)
$\overline{M}_1 = p_1 + p_2$ | $|T(K_2^{\overline{M}_1})| = |\{t_1 - t_4\}| = 4$
t_2
t_5
$\overline{M}_7 = p_2 + p_4$ | $|T(K_2^{\overline{M}_7})| = 0$
$\overline{M}_3 = p_1 + p_7$ | $|T(K_2^{\overline{M}_3})| = |\{t_1, t_2\}| = 2$
t_5
t_2
$\overline{M}_8 = p_4 + p_7$ | $|T(K_2^{\overline{M}_8})| = 0$
(d)
$\overline{M}_1 = p_1 + p_2$ | $|T(K_2^{\overline{M}_1})| = |\{t_1 - t_4\}| = 4$
t_2
t_6
$\overline{M}_7 = p_2 + p_4$ | $|T(K_2^{\overline{M}_7})| = 0$
$\overline{M}_5 = p_1 + p_8$ | $|T(K_2^{\overline{M}_5})| = |\{t_1, t_2\}| = 2$
t_6
t_2
$\overline{M}_9 = p_4 + p_8$ | $|T(K_2^{\overline{M}_9})| = 0$
(e)

FIGURE 5: The partial reachability graph of a CDC $(\overline{N}_2, \overline{M}_1)$ with $|T(K_2)| = 4$ by concurrently firing transitions in $D_1^{\overline{M}_1} = \{t_1, t_5\}$, $D_2^{\overline{M}_1} = \{t_1, t_6\}$, $D_3^{\overline{M}_1} = \{t_2, t_5\}$, and $D_4^{\overline{M}_1} = \{t_2, t_6\}$.

conflicting transitions $t_1 - t_4$ are executed, where $\overline{M}_3 = p_1 + p_7$. In this case, we can say that the disappearance of the effective conflict $K_2^{\overline{M}_1}$ occurs if the confusion fires transition sequence $\sigma_2 = t_5 t_1$.

The concurrent firing of the transitions in other sets $D_2^{\overline{M}_1}$, $D_3^{\overline{M}_1}$, and $D_4^{\overline{M}_1}$ can be analyzed by the similar procedures of the analysis for transitions t_1 and t_5 in $D_1^{\overline{M}_1}$. Specifically, the disappearance of the effective conflict $K_2^{\overline{M}_1}$ occurs if the confusion fires transition sequence $\sigma_3 = t_6 t_1$, $\sigma_4 = t_5 t_2$, or $\sigma_5 = t_6 t_2$.

4. Detection of Confusions in WF-Systems

Conflicts and concurrency are used to control the option processes and to keep a high-performance operation of a WF-system. However, confusions may lead to the incomplete and nondeterministic conflicting behavior.

This section deals with the detection problem of confusions in WF-systems. A confusion is tied to a special structure and a marking. Hence, the detection method is divided into two steps. First, the structure $\overline{N}$ of a confusion $(\overline{N}, \overline{M})$ in a WF-system is detected. Then, whether there exists a marking $\overline{M}$ such that the confusion occurs is decided by a method of integer linear programming (ILP), which does not require computing the reachability graph of the WF-system.

Property 1. If $\overline{N}$ is a CIC subnet in a WF-net N, then $\exists t_i, t_j, t_k \in \overline{T}, i \neq j \neq k$ such that $({}^\bullet t_i \cup t_i^\bullet) \cap ({}^\bullet t_j \cup t_j^\bullet) = \emptyset \wedge t_k \in (t_j^\bullet)^\bullet \wedge {}^\bullet t_i \cap {}^\bullet t_k \neq \emptyset$ is true. If $\overline{N}$ is a CDC subnet in a WF-net N, then $\exists t_i, t_j \in T$, $i \neq j$ such that $({}^\bullet t_i \cup t_i^\bullet) \cap ({}^\bullet t_j \cup t_j^\bullet) = \emptyset \wedge ({}^\bullet t_i)^\bullet \cap ({}^\bullet t_j)^\bullet \neq \emptyset$ is true.

Property 1 depicts the structural characteristics of a CIC and a CDC described in Definition 9. Algorithm 1 is developed to capture all confusion subnets in a WF-net according to Property 1. Let $T(K,t) := \{t' \mid t' \in T \wedge {}^\bullet t \cap {}^\bullet t'\}$ in Algorithm 1 denote the conflicting transition set containing t.

In a WF-system, the captured confusion subnets cannot perfectly describe the phenomenon of confusions since they are not related to markings. It does not mean that a confusion $(\overline{N}, \overline{M})$ necessarily exists if there is a confusion subnet $\overline{N}$ in a WF-system. Hence, we need to decide whether there exists a marking $\overline{M}$ such that confusion $(\overline{N}, \overline{M})$ occurs.

Theorem 10. *Let (N, M_{in}) be a sound WF-system. Let $I_1, I_2, \ldots,$ and I_n denote all the minimal P-invariants computed by $I^T[N] = \mathbf{0}^T$. Then (N, M_{in}) is covered by $I_1, I_2, \ldots,$ and I_n and $\forall j \in \{1, 2, \ldots, n\}$; $\Sigma_{p \in \|I_j\|} M(p) = M_{in}(p_{in}) = 1$ is true.*

Proof. According to Definition 1, we have $\forall p_i \in P$; p_i belongs to an elementary directed path from p_{in} to p_{out} in N. Any elementary directed path in a WF-system (N, M_{in}) can deduce a minimal P-invariant. Hence, (N, M_{in}) is covered by $I_1, I_2, \ldots,$ and I_n if there are total n minimal P-invariants. A P-invariant I_j holds for $I_j^T \cdot M = I_j^T \cdot M_{\text{in}} = 1$. Hence, we have $\Sigma_{p \in \|I_j\|} M(p) = M_{\text{in}}(p_{\text{in}}) = 1$. □

Algorithm 2 is proposed according to Theorem 10, which is referred to find the places that may lead to the disappearance of an effective conflict in a confusion if they are marked. Then, an ILP is developed to decide whether a marking exists such that all these places are marked. If the result of the ILP equals the number of the places, an existing confusion in a WF-system can be finally determined.

Let (N, M_{in}) be a WF-system and Q_{ci} and Q_{cd} denote the sets of its CIC and CDC subnets obtained by Algorithm 1, respectively. Suppose that $\overline{N} = (\overline{P}, \overline{T}, \overline{F}) \in Q_{ci}$ is a CIC subnet in (N, M_{in}), where $T(K)$ is the transition set of a structural conflict K in N, $\Omega_1 = \overline{T} \setminus T(K)$ is a set containing all nonconflicting transitions in $\overline{T}$, and $\Omega_2 = T(K) \setminus (T(K) \cap (\Omega_1^\bullet)^\bullet)$ is a transition set with respect to K in $\overline{T}$. If there exists a marking $\overline{M} \prec M \in R(N, M_{\text{in}})$ with $\Sigma_{p \in {}^\bullet(\Omega_1 \cup \Omega_2)} M(p) = |{}^\bullet(\Omega_1 \cup \Omega_2)|$, that is, $\forall p \in {}^\bullet(\Omega_1 \cup \Omega_2)$, p is marked at M, we can decide that $(\overline{N}, \overline{M})$ is a CIC that can occur in the WF-system (N, M_{in}). Similarly, for a CDC subnet $\overline{N}$ in a WF-system (N, M_{in}), if there exists a marking $\overline{M} \prec M \in R(N, M_{\text{in}})$ with $\Sigma_{p \in {}^\bullet \overline{T}} M(p) = |{}^\bullet \overline{T}|$, we can decide that $(\overline{N}, \overline{M})$ is a CDC in the WF-system (N, M_{in}). The decision processes are formulated in Algorithm 2.

Two WF-systems (N, M_{in}) and (N', M'_{in}) are used to illustrate the proposed detection policy of confusions, which are shown in Figures 6(a) and 6(b), respectively. The WF-system (N, M_{in}) is sound and another WF-system (N', M'_{in}) is sound and well structured, where anyone of them contains a CIC.

If Algorithm 1 is applied in WF-net N, the CIC subnet $\overline{N}$ shown in Figure 6(c) can be obtained, where $t_i := t_2, t_j := t_3$, $t_k := t_5$, and $({}^\bullet t_2 \cup t_2^\bullet) \cap ({}^\bullet t_3 \cup t_3^\bullet) = \emptyset \wedge t_5 \in (t_3^\bullet)^\bullet \wedge {}^\bullet t_2 \cap {}^\bullet t_5 \neq \emptyset$ is true. Hence, the sets $\overline{P}$, $\overline{T}$, and $\overline{F}$ in $\overline{N}$ can be computed; that is,

(1) $\overline{P} := {}^\bullet T(K, t_5) \cup T(K, t_5)^\bullet \cup {}^\bullet t_3 \cup (t_3^\bullet \cap {}^\bullet T(K, t_5)) = \{p_1, p_4\} \cup \{p_3, p_{\text{out}}\} \cup \{p_2\} \cup \{p_4\} = \{p_1 - p_4, p_{\text{out}}\}$;

(2) $\overline{T} := T(K, t_5) \cup \{t_3\} = \{t_2, t_5\} \cup \{t_3\} = \{t_2, t_3, t_5\}$;

(3) $\overline{F} := F \cap ((\overline{P} \times \overline{T}) \cup (\overline{T} \times \overline{P}))$.

Algorithm 2 is applied to decide whether there exists a reachable marking $\overline{M}$ in (N, M_{in}) such that $(\overline{N}, \overline{M})$ is a CIC. First, the supports of three minimal P-invariants $I_1 - I_3$, that is, $\|I_1\| = \{p_{\text{in}}, p_1, p_3, p_{\text{out}}\}$, $\|I_2\| = \{p_{\text{in}}, p_1, p_{\text{out}}\}$, and $\|I_3\| = \{p_{\text{in}}, p_2, p_4, p_{\text{out}}\}$, are obtained by computing $[N]^T \cdot I = \mathbf{0}$. Then, the ILP depicted in line 8 of Algorithm 2 can be implemented and returns a value $v = 2$. It implies that there exists a marking in WF-system (N, M_{in}) such that places p_1 and p_2 in $\overline{N}$ are marked. In other words, a confusion $(\overline{N}, \overline{M})$ with $\overline{M} = p_1 + p_2$ is detected, which will occur in the WF-system (N, M_{in}). Similarly, in WF-system (N', M'_{in}), the confusion $(\overline{N'}, \overline{M'})$ shown in Figure 6(d) can be detected by the procedures of Algorithms 1 and 2. These examples

```
Input: WF-net $N = (P, T, F)$.
Output: The set $Q_{\text{ci}}$ of CIC subnets and the set $Q_{\text{cd}}$ of CDC subnets.
$Q_{\text{ci}} := \emptyset, Q_{\text{cd}} := \emptyset, h := 0$.
for each $t_i, t_j, t_k \in T, i \neq j \neq k$ do
    if $({}^{\bullet}t_i \cup t_i^{\bullet}) \cap ({}^{\bullet}t_j \cup t_j^{\bullet}) = \emptyset \wedge t_k \in (t_j^{\bullet})^{\bullet} \wedge {}^{\bullet}t_i \cap {}^{\bullet}t_k \neq \emptyset$ then
        Capture a CIC subnet $\overline{N}_h = (\overline{P}_h, \overline{T}_h, \overline{F}_h)$, where
            (1) $\overline{P}_h :={}^{\bullet} T(K, t_k) \cup T(K, t_k)^{\bullet} \cup {}^{\bullet} t_j \cup (t_j^{\bullet} \cap {}^{\bullet} T(K, t_k))$,
            (2) $\overline{T}_h := T(K, t_k) \cup \{t_j\}$, and
            (3) $\overline{F}_h := F \cap ((\overline{P}_h \times \overline{T}_h) \cup (\overline{T}_h \times \overline{P}_h))$.
        if $\overline{N}_h \notin Q_{\text{ci}}$ then
            $h := h + 1, Q_{\text{ci}} := Q_{\text{ci}} \cup \overline{N}_h$.
        else
            $Q_{\text{ci}} := Q_{\text{ci}}$.
        end if
    end if
end for
for each $t_i, t_j \in T, i \neq j$ do
    If $({}^{\bullet}t_i \cup t_i^{\bullet}) \cap ({}^{\bullet}t_j \cup t_j^{\bullet}) = \emptyset \wedge ({}^{\bullet}t_i)^{\bullet} \cap ({}^{\bullet}t_j)^{\bullet} \neq \emptyset$
    then
        Capture a CDC subnet $\overline{N}_h = (\overline{P}_h, \overline{T}_h, \overline{F}_h)$, where
            (1) $\overline{P}_h : ={}^{\bullet} T(K, t_i) \cup T(K, t_i)^{\bullet} \cup {}^{\bullet} T(K, t_j) \cup T(K, t_j)^{\bullet}$,
            (2) $\overline{T}_h := T(K, t_i) \cup T(K, t_j)$, and
            (3) $\overline{F}_h := F \cap ((\overline{P}_h \times \overline{T}_h) \cup (\overline{T}_h \times \overline{P}_h))$.
        if $\overline{N}_h \notin Q_{\text{cd}}$ then
            $h := h + 1, Q_{\text{cd}} := Q_{\text{cd}} \cup \overline{N}_h$.
        else
            $Q_{\text{cd}} := Q_{\text{cd}}$.
        end if
    end if
end for
Output: $Q_{\text{ci}}$ and $Q_{\text{cd}}$.
```

Algorithm 1: Detection of confusion subnets in WF-net N.

show that the detection method developed in this paper can effectively detect confusions in WF-systems.

The WF-system (N'', M''_{in}) shown in Figure 7(a) depicts a real-world case of complaint processes, which is reported in [9]. There are four structural conflicts $K_1 = \langle p_3, \{t_3, t_4\} \rangle$, $K_2 = \langle p_4, \{t_8, t_{13}\} \rangle$, $K_3 = \langle p_6, \{t_6, t_7\} \rangle$, and $K_4 = \langle p_{11}, \{t_{11}, t_{12}\} \rangle$, which implement the logical judgments of WF-system (N'', M''_{in}) such that a complaint case can be complete. According to the study in this paper, the prerequisite of implementing reasonable logical judgment is that the WF-system (N'', M''_{in}) is confusion-free. Hence, Algorithms 1 and 2 are considered in (N'', M''_{in}). First, the confusion subnet N'' shown in Figure 7(b) can be detected by Algorithm 1. However, a confusion with respect to N'' will not occur in the WF-system (N'', M''_{in}) since the ILP in Algorithm 2 returns the value $v = 2$. It is shown that no reachable marking exists in the WF-system such that all the places p_4, p_6, and p_7 in N'' are marked. Finally, a conclusion that the complaint process will not cause a confusion and its WF-system is confusion-free is obtained. The WF-system can be correctly used to model and analyze complaint cases without considering the disappearance of any conflict.

5. Impact of Confusions on the Properties of WF-Systems

The algorithms developed in Section 4 show a policy to detect confusions in a WF-system. In the following, the impact of confusions on the properties of WF-systems is specified.

Property 2. A sound and well-structured WF-net (N, M_{in}) with cycles may contain confusions.

Proof. The example shown in Figure 6(b) that is a sound and well-structured with a cycle. According to Algorithms 1 and 2, a CIC shown in Figure 6(d) can be obtained. Hence, a sound and well-structured WF-net (N, M_{in}) with cycles may contain confusions. □

Property 3. An acyclic sound WF-system (N, M_{in}) is not well structured if (N, M_{in}) contains a confusion $(\overline{N}, \overline{M})$ with $\overline{M} \prec M \in R(N, M_{\text{in}})$.

```
Input: WF-system $(N, M_{\text{in}})$, $Q_{\text{ci}}$, and $Q_{\text{cd}}$.
Output: The set $\widehat{Q}_{\text{ci}}$ of CICs and the set $\widehat{Q}_{\text{cd}}$ of CDCs.
$\widehat{Q}_{\text{ci}} := \emptyset$, $\widehat{Q}_{\text{cd}} := \emptyset$,
for each $\overline{N}_h \in Q_{\text{ci}}$ do
    Compute $[N]^T \cdot I = \mathbf{0}$ to obtain all minimal $P$-invariants $I_1, I_2, \ldots$, and $I_n$.
    Find $K = \langle p, T(K) \rangle$ in $\overline{N}_h$.
    $\Omega_1 := \overline{T} \setminus T(K)$, $\Omega_2 := T(K) \setminus \left(T(K) \cap (\Omega_1^{\bullet})^{\bullet}\right)$.
    Compute the value $v$ of following ILP:
        $\left( \begin{array}{c} \max \Sigma_{p \in {}^{\bullet}(\Omega_1 \cup \Omega_2)} M(p) \\ \text{s.t.}: \Sigma_{p \in ||I_1||} M(p) = M_{\text{in}}(p_{\text{in}}) = 1 \\ \Sigma_{p \in ||I_2||} M(p) = M_{\text{in}}(p_{\text{in}}) = 1, \\ \vdots \\ \Sigma_{p \in ||I_n||} M(p) = M_{\text{in}}(p_{\text{in}}) = 1. \end{array} \right)$
    if $v = \left|{}^{\bullet}(\Omega_1 \cup \Omega_2)\right|$ then
        $\overline{M} := \Sigma_{p \in {}^{\bullet}(\Omega_1 \cup \Omega_2)} M(p)\, p$.
        $\widehat{Q}_{\text{ci}} := \widehat{Q}_{\text{ci}} \cup \left(\overline{N}_h, \overline{M}\right)$.
    else
        $\widehat{Q}_{\text{ci}} := \widehat{Q}_{\text{ci}}$.
    end if
end for
for each $\overline{N}_h \in Q_{\text{cd}}$ do
    Compute $[N]^T \cdot I = \mathbf{0}$ to obtain all minimal $P$-invariants $I_1, I_2, \ldots$, and $I_n$.
    Compute the value $v$ of following ILP:
        $\left( \begin{array}{c} \max \Sigma_{p \in {}^{\bullet}\overline{T}} M(p) \\ \text{s.t.}: \Sigma_{p \in ||I_1||} M(p) = M_{\text{in}}(p_{\text{in}}) = 1 \\ \Sigma_{p \in ||I_2||} M(p) = M_{\text{in}}(p_{\text{in}}) = 1, \\ \vdots \\ \Sigma_{p \in ||I_n||} M(p) = M_{\text{in}}(p_{\text{in}}) = 1. \end{array} \right)$
    if $v = \left|{}^{\bullet}\overline{T}\right|$ then
        $\overline{M} := \Sigma_{p \in {}^{\bullet}\overline{T}} M(p)\, p$.
        $\widehat{Q}_{\text{cd}} := \widehat{Q}_{\text{cd}} \cup \left(\overline{N}_h, \overline{M}\right)$.
    else
        $\widehat{Q}_{\text{cd}} := \widehat{Q}_{\text{cd}}$.
    end if
end for
Output: $\widehat{Q}_{\text{ci}}$ and $\widehat{Q}_{\text{cd}}$.
```

Algorithm 2: Detection of confusions by the ILP in WF-system (N, M_{in}).

Proof. There are two cases only for confusion subnet $\overline{N}$.

Case (1). If $\overline{N}$ is a CIC subnet, there must be three differentiable transitions t_i, t_j, and t_k holding for $({}^{\bullet}t_i \cup t_i^{\bullet}) \cap ({}^{\bullet}t_j \cup t_j^{\bullet}) = \emptyset$, $t_k \in (t_j^{\bullet})^{\bullet}$, and ${}^{\bullet}t_i \cap {}^{\bullet}t_k \neq \emptyset$. According to Definitions 1 and 3, and ${}^{\bullet}t_i \cap {}^{\bullet}t_j = \emptyset$, there must be some places and a transition $t \in \overline{T}$ modeling an AND-split such that two different elementary paths $C_1 = p_{\text{in}}, \ldots, t, \ldots, p_a, \ldots, p_{\text{out}}$ and $C_2 = p_{\text{in}}, \ldots, t, \ldots, p_b, \ldots, p_{\text{out}}$ in N exist, where $p_a \in {}^{\bullet}t_i$, $p_b \in {}^{\bullet}t_j$, and $p_a \neq p_b$. Otherwise, if $t \notin \overline{T}$, $\nexists \overline{M} \prec M \in R(N, M_{\text{in}})$ holding for $\overline{M}[t_i\rangle$ and $\overline{M}[t_j\rangle$. In other words, there is a dead transition between t_i and t_j in the extended WF-system $(\widetilde{N}, M_{\text{in}})$, which violates the sound property. Hence, transition t necessarily exists.

(N, M_{in}) is sound and transition t splits a token into multiple tokens. Hence, there must be some transitions and places with $p \in \overline{P}$ modeling an AND-join such that the tokens in different places can converge to p. We can deduce that there exist two different elementary paths $C_1 = p_{\text{in}}, \ldots, t, \ldots, p_a, \ldots, p, \ldots, p_{\text{out}}$ and $C_2 = p_{\text{in}}, \ldots, t, \ldots, p_b, \ldots, p, \ldots, p_{\text{out}}$ in N. Otherwise, if $p \notin \overline{P}$, we can conclude that $\exists M \in R(N, M_{in}) \wedge M \geq M_{\text{out}} \nRightarrow M = M_{out}$. In other words, there are tokens in the places belonging to $\overline{P} \setminus \{p_{\text{in}} \cup p_{\text{out}}\}$ at the terminal state in (N, M_{in}), which also violates the sound property. According to C_1 and C_2, we have two different elementary paths $C_1' = t, \ldots, p_a, \ldots, p$ and $C_2' = t, \ldots, p_b, \ldots, p$ owing to $p_a \neq p_b$. However, $\alpha(C_1') \cap \alpha(C_2') = \{t, p\}$ cannot deduce $C_1' = C_2'$. According to Definition 4, (N, M_{in}) is not well structured.

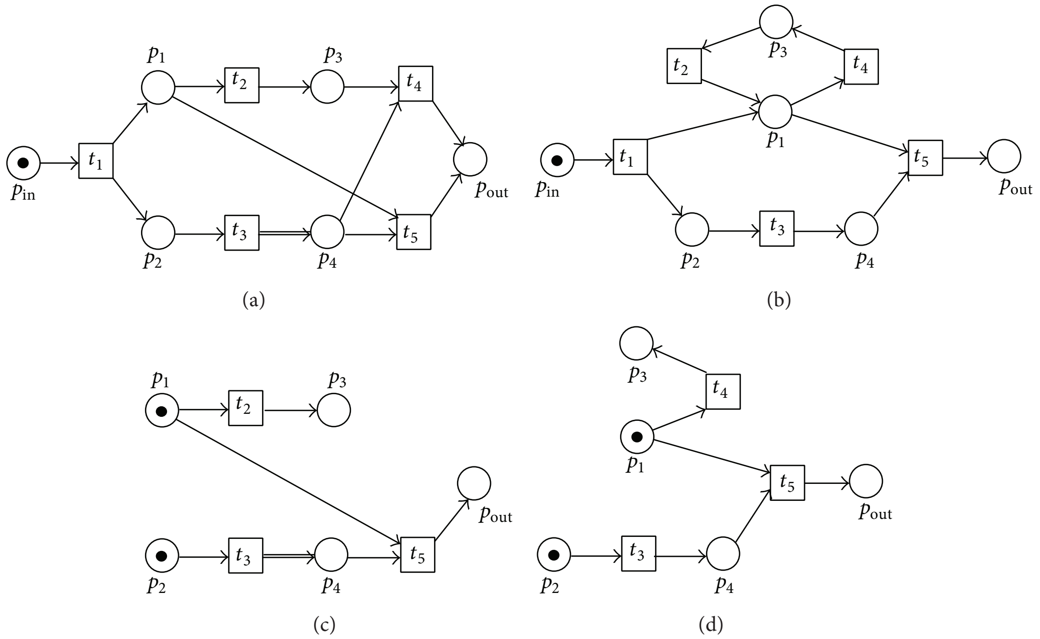


FIGURE 6: (a) A sound WF-system (N, M_{in}), (b) a sound and well-structured WF-system (N', M'_{in}), (c) a CIC $(\overline{N}, \overline{M})$ in (N, M_{in}), and (d) a CIC $(\overline{N}', \overline{M}')$ in (N', M'_{in}).

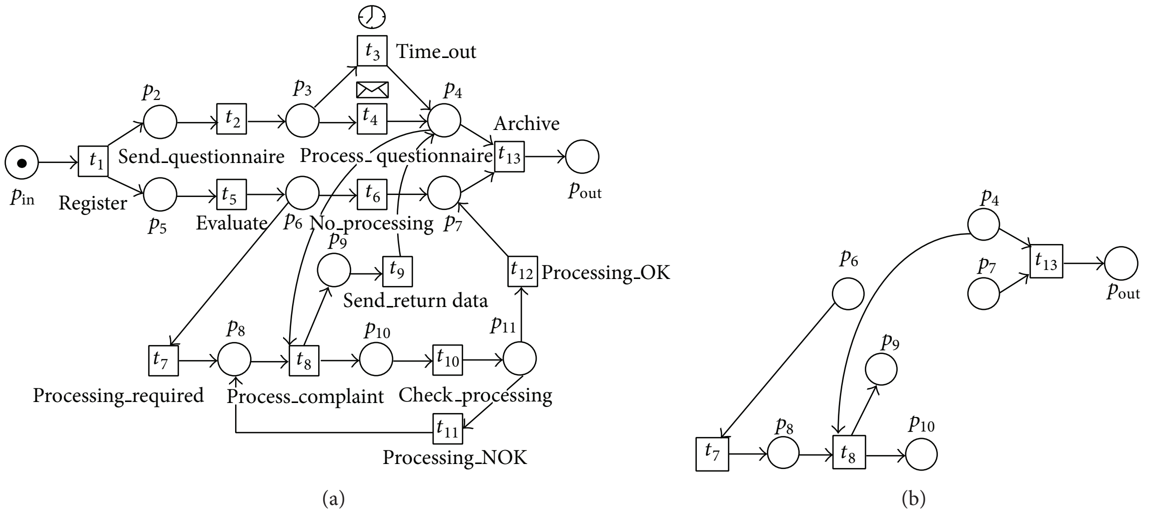


FIGURE 7: (a) A real-world WF-system (N'', M''_{in}) of complaint processes and (b) a CIC subnet N'' in (N'', M''_{in}).

Case (2). If $\overline{N}$ is a CDC subnet, there must be three differentiable transitions t_i, t_j, and t_k holding for $({}^\bullet t_i \cup t_i^\bullet) \cap ({}^\bullet t_j \cup t_j^\bullet) = \emptyset$, $({}^\bullet t_i)^\bullet \cap ({}^\bullet t_j)^\bullet \neq \emptyset$, and $t_k \in ({}^\bullet t_i)^\bullet \cap ({}^\bullet t_j)^\bullet$. WF-system (N, M_{in}) is sound. Hence, there exists a marking $\overline{M}$ with respect to $\overline{M} \prec M \in R(N, M_{\text{in}})$ holding for $\overline{M}[t_i\rangle$ and $\overline{M}[t_j\rangle$. Otherwise, if $\overline{M}$ does not exist, transition t_k will be dead owing to $t_k \in ({}^\bullet t_i)^\bullet \cap ({}^\bullet t_j)^\bullet \neq \emptyset$. According to Definition 1, ${}^\bullet t_i \cap {}^\bullet t_j = \emptyset$ and $t_i^\bullet \cap t_j^\bullet = \emptyset$, there must be some places and a transition $t \in \overline{T}$ modeling an AND-split and also exist some transitions and places with $p \in \overline{P}$ modeling an AND-join to product two different elementary paths $C_1 = p_{\text{in}}, \ldots, t, \ldots, p_a, \ldots, p, \ldots, p_{\text{out}}$ and $C_2 = p_{\text{in}}, \ldots, t, \ldots, p_b, \ldots, p, \ldots, p_{\text{out}}$, where $p_a \in {}^\bullet t_i$, $p_b \in {}^\bullet t_j$, and $p_a \neq p_b$. Similarly, $C'_1 = t, \ldots, p_a, \ldots, p$ and $C'_2 = t, \ldots, p_b, \ldots, p$ are also two different elementary paths. According to Definition 4, (N, M_{in}) is not well structured. □

According to Property 3, confusions can affect the property "well-structured" of an acyclic sound WF-system. As an example, the acyclic sound WF-system shown in Figure 6(a) is not well structured owing to the existence of a CIC.

6. Conclusions and Future Work

Confusion detection problems in WF-systems are tackled in this paper. First, the work explores CICs and CDCs in a WF-system by formalizing them as a class of marked subnets with respect to the limitation of conflicts and concurrency. The disappearance of conflicts in a confusion can be described in such marked subnets, which motivates us to develop confusion detection methods proposed in this paper.

Second, a policy is developed to capture confusions, in which a detection algorithm is proposed to find confusion subnets in a WF-system and a decision algorithm based on ILP is proposed to decide whether the found subnets will eventually cause confusions at a reachable marking. The policy avoids computing the reachability graph of a WF-system. Two examples and a real-world case of WF-systems reported by van der Aalst [9] are used to illustrate the detection of confusions. The results show that the detection method is feasible and efficient.

Third, the impact of confusions on the properties of WF-systems is specified. A conclusion is obtained; that is, if an acyclic WF-system contains a confusion, it is certainly not well structured. Future work attempts to find the avoidance policy of confusions in WF-systems by using timing constraints in [3] or supervisory control in [6].

Conflict of Interests

The first author and any of the coauthors do not have any conflict of interests regarding this paper.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant no. 61374068, the Fundamental Research Funds for the Central Universities under Grant no. JB-ZR1130, and the National Natural Science Foundation of China under Grant no. 61304051.

References

[1] W. Tan and M. C. Zhou, *Business and Scientific Workflows: A Service-Oriented Approach*, Wiley-IEEE Press, 2013.

[2] W. M. P. van der Aalst, “Verification of workflow nets,” in *Application and Theory of Petri Nets*, P. Azma and G. Balbo, Eds., vol. 1248 of *Lecture Notes in Computer Science*, pp. 407–426, Springer, Berlin, Germany, 1997.

[3] J. Q. Li, Y. S. Fan, and M. C. Zhou, “Timing constraint workflow nets for workflow analysis,” *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 33, no. 2, pp. 179–193, 2003.

[4] J. Q. Li, Y. S. Fan, and M. C. Zhou, “Performance modeling and analysis of workflow,” *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 34, no. 2, pp. 229–242, 2004.

[5] T. Murata, “Petri nets: properties, analysis and applications,” *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[6] Z. W. Li and M. C. Zhou, *Deadlock Resolution in Automated Manufacturing Systems: A Novel Petri Net Approach*, Springer, London, UK, 2009.

[7] Z. W. Li, N. Q. Wu, and M. C. Zhou, “Deadlock control of automated manufacturing systems based on Petri nets—a literature review,” *IEEE Transactions on Systems, Man and Cybernetics C: Applications and Reviews*, vol. 42, no. 4, pp. 437–462, 2012.

[8] M. W. P. van der Aalst and K. Hee, *Workflow Management: Models, Methods, and Systems*, The MIT Press, Cambridge, UK, 2004.

[9] M. W. P. van der Aalst, “Workflow verification: finding control-flow errors using Petri-net-based tech niques,” in *Business Process Management: Models, Techniques, and Empirical Studies*, W. van der Aalst, J. Desel, and A. Oberweis, Eds., vol. 1806 of *Lecture Notes in Computer Science*, pp. 19–128, Springer, Berlin, Germany, 2000.

[10] P. C. Xiong, Y. S. Fan, and M. C. Zhou, “A Petri net approach to analysis and composition of web services,” *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 40, no. 2, pp. 376–387, 2010.

[11] X. T. Li, Y. S. Fan, Q. Z. Sheng, Z. Maamar, and H. W. Zhu, “A Petri net approach to analyzing behavioral compatibility and similarity of web services,” *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 41, no. 3, pp. 510–521, 2011.

[12] Y. F. Chen and Z. W. Li, “Design of a maximally permissive liveness-enforcing supervisor with a compressed supervisory structure for flexible manufacturing systems,” *Automatica*, vol. 47, no. 5, pp. 1028–1034, 2011.

[13] Z. W. Li and M. C. Zhou, “Two-stage method for synthesizing liveness-enforcing supervisors for flexible manufacturing systems using Petri nets,” *IEEE Transactions on Industrial Informatics*, vol. 2, no. 4, pp. 313–325, 2006.

[14] G. Rozenberg and P. S. Thiagarajan, “Petri nets: basic notions, structure, behaviour,” in *Current Trends in Concurrency: Overviews and Tutorials*, J. de Bakker, W. de Roever, and G. Rozenberg, Eds., vol. 224 of *Lecture Notes in Computer Science*, pp. 585–668, Springer, Berlin, Germany, 1986.

[15] G. Rozenberg and J. Engelfriet, “Elementary net systems,” in *Lectures on Petri Nets I: Basic Models*, W. Reisig and G. Rozenberg, Eds., vol. 1491 of *Lecture Notes in Computer Science*, pp. 12–121, Springer, Berlin, Germany, 1998.

[16] E. Smith, “On the border of causality: contact and confusion,” *Theoretical Computer Science*, vol. 153, no. 1-2, pp. 245–270, 1996.

[17] S. Haar, “Clusters, confusion and unfoldings,” *Fundamenta Informaticae*, vol. 47, no. 3-4, pp. 259–270, 2001.

[18] C. Bolton, “Adding conflict and confusion to CSP,” in *Proceedings of the International Symposium of Formal Methods Europe (FM ’05)*, J. Fitzgerald, J. Hayes, and A. Tarlecki, Eds., vol. 3582 of *Lecture Notes in Computer Science*, pp. 205–220, Springer, July 2005.

[19] C. Bolton, “Capturing conflict and confusion in CSP,” in *Proceedings of the 6th International Conference on Integrated Formal Methods*, J. Davies and J. Gibbons, Eds., vol. 4591 of *Lecture Notes in Computer Science*, pp. 413–438, Springer, 2007.

[20] X.-L. Chen, Z.-W. Li, A. M. Al-Ahmari, A. M. El-Tamimi, and E. S. Abouel Nasr, “Confusion diagnosis and control of descrete event systems using synchronized Petri nets,” *Asian Journal of Control*, vol. 15, no. 6, pp. 1736–1751, 2013.

[21] J. Katoen, “GSPNs revisited: simple semantics and new analysis algorithms,” in *Proceedings of the International Conference on Application of Concurrency to System Design*, pp. 6–11, Hamburg, Germany, 2012.

[22] M. Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis, *Modelling with Generalized Stochastic Petri Nets*, John Wiley & Sons, New York, NY, USA, 1995.

[23] S. Haar, "Occurrence net logics," *Fundamenta Informaticae*, vol. 43, no. 1–4, pp. 105–127, 2000.

[24] Y. Y. Du, C. J. Jiang, M. C. Zhou, and Y. Fu, "Modeling and monitoring of E-commerce workflows," *Information Sciences*, vol. 179, no. 7, pp. 995–1006, 2009.

[25] Y. F. Chen, Z. W. Li, M. Khalgui, and O. Mosbahi, "Design of a maximally permissive liveness-enforcing Petri net supervisor for flexible manufacturing systems," *IEEE Transactions on Automation Science and Engineering*, vol. 8, no. 2, pp. 374–393, 2011.

[26] Z. W. Li and M. C. Zhou, "Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 34, no. 1, pp. 38–51, 2004.

[27] Z. W. Li and M. C. Zhou, "Control of elementary and dependent siphons in Petri nets and their application," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 38, no. 1, pp. 133–148, 2008.

[28] Z. W. Li, M. C. Zhou, and N. Q. Wu, "A survey and comparison of Petri net-based deadlock prevention policies for flexible manufacturing systems," *IEEE Transactions on Systems, Man and Cybernetics C: Applications and Reviews*, vol. 38, no. 2, pp. 173–188, 2008.

[29] Z. W. Li, M. C. Zhou, and M. D. Jeng, "A maximally permissive deadlock prevention policy for FMS based on Petri net siphon control and the theory of regions," *IEEE Transactions on Automation Science and Engineering*, vol. 5, no. 1, pp. 182–188, 2008.

[30] Z. W. Li and M. Zhao, "On controllability of dependent siphons for deadlock prevention in generalized Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 38, no. 2, pp. 369–384, 2008.

[31] Z. W. Li, G. Y. Liu, H.-M. Hanisch, and M. C. Zhou, "Deadlock prevention based on structure reuse of Petri net supervisors for flexible manufacturing systems," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 42, no. 1, pp. 178–191, 2012.

[32] R. David and H. Alla, *Discrete, Continuous, and Hybrid Petri Nets*, Springer, Berlin, Germany, 2005.

Advances in Operations Research
Advances in Decision Sciences
Journal of Applied Mathematics
Algebra
Journal of Probability and Statistics
The Scientific World Journal
International Journal of Differential Equations
International Journal of Combinatorics
Advances in Mathematical Physics
Journal of Complex Analysis
Journal of Mathematics
Mathematical Problems in Engineering
Abstract and Applied Analysis
Discrete Dynamics in Nature and Society
International Journal of Mathematics and Mathematical Sciences
Journal of Discrete Mathematics
Journal of Function Spaces
International Journal of Stochastic Analysis
Journal of Optimization