LETTER *Special Section on Formal Approach*

Refactoring Problem of Acyclic Extended Free-Choice Workflow Nets to Acyclic Well-Structured Workflow Nets

Shingo YAMAGUCHI[†a)], ***Senior Member***

SUMMARY A workflow net (WF-net for short) is a Petri net which represents a workflow. There are two important subclasses of WF-nets: extended free-choice (EFC for short) and well-structured (WS for short). It is known that most actual workflows can be modeled as EFC WF-nets; Acyclic WS is a subclass of acyclic EFC but has more analysis methods. An acyclic EFC WF-net may be transformed to an acyclic WS WF-net without changing the external behavior of the net. We name such a transformation *Acyclic EFC WF-net refactoring*. We give a formal definition of acyclic EFC WF-net refactoring problem. We also give a necessary condition and a sufficient condition for solving the problem. Those conditions can be checked in polynomial time. These result in the enhancement of the analysis power of acyclic EFC WF-nets.

key words: *workflow net, refactoring, well-structured, soundness, branching bisimilarity*

1. Introduction

A workflow net (WF-net for short) [1] is a Petri net [2] which represents a workflow. There are two important subclasses of WF-nets: extended free-choice (EFC for short) and well-structured (WS for short). It is known that most actual workflows can be modeled as EFC WF-nets; Acyclic WS is a subclass of acyclic EFC but has more analysis methods, e.g. heuristic computation of parallel degree [3].

In software engineering, code refactoring [4] has been attracting a great deal of attention in order to reduce the complexity of code. Code refactoring is to transform a source code to a new form without changing its external behavior. We try to introduce the concept of refactoring to the analysis of WF-nets. An acyclic EFC WF-net may be transformed to an acyclic WS WF-net without changing its external behavior. We name such a transformation *acyclic EFC WF-net refactoring*. If a given acyclic EFC WF-net is refactored to an acyclic WS WF-net, we can use the analysis methods of WS WF-nets to analyze the EFC WF-net. This results in the enhancement of the analysis power of EFC WF-nets.

In this paper, we give a formal definition of acyclic EFC WF-net refactoring problem. Next we give a necessary condition and a sufficient condition for solving the problem. We also show that those conditions can be checked in polynomial time.

Manuscript received July 16, 2011.
Manuscript revised November 7, 2011.

[†]The author is with the Graduate School of Science and Engineering, Yamaguchi University, Ube-shi, 755–8611 Japan.
a) E-mail: shingo@yamaguchi-u.ac.jp

DOI: 10.1587/transinf.E95.D.1375

2. Workflow Nets and Properties

A (labeled) WF-net is a labeled Petri net which represents a workflow. A labeled Petri net is a four tuple $N{=}(P, T, A, \ell)$. P, T, and A ($\subseteq(P{\times}T)\cup(T{\times}P)$) are finite sets of *places*, *transitions*, and *arcs*, respectively. Each transition of a WF-net represents an action. Some actions can be observed, others cannot. The former and the latter are called *external* and *internal*, respectively. Actions are identified by label. Internal actions are labeled as a designated label τ. $\ell : T{\rightarrow}\mathcal{A}\cup\{\tau\}$ is a labelling function of transitions, where $\mathcal{A}$ denotes the set of all possible external labels.

Definition 1 (WF-net [1]): A labeled Petri net $N{=}(P, T, A, \ell)$ is a (labeled) WF-net iff (i) N has a single source place p_I ($\overset{N}{\bullet}p_I{=}\emptyset$ and $\forall p{\in}(P{-}\{p_I\}) : \overset{N}{\bullet}p{\neq}\emptyset$) and a single sink place p_O ($p_O\overset{N}{\bullet}{=}\emptyset$ and $\forall p{\in}(P{-}\{p_O\}) : p\overset{N}{\bullet}{\neq}\emptyset$), where for a node x ($\in(P{\cup}T)$), $\overset{N}{\bullet}x$ and $x\overset{N}{\bullet}$ denote $\{y|(y, x){\in}A\}$ and $\{y|(x, y){\in}A\}$, respectively; and (ii) every place or transition is on a path from p_I to p_O. □

Let $N{=}(P, T, A, \ell)$ be a WF-net. We represent a marking of N as a bag over P. A marking is denoted by $M{=}[p^{M(p)}|p{\in}P, M(p){>}0]$, where $M(p)$ denotes the number of tokens in p. Let M_X and M_Y be markings. $M_X{=}M_Y$ denotes that $\forall p{\in}P : M_X(p){=}M_Y(p)$. $M_X{\geq}M_Y$ denotes that $\forall p{\in}P : M_X(p){\geq}M_Y(p)$. A transition t is said to be firable in a marking M if $M{\geq}\overset{N}{\bullet}t$. Firing t in M results in a new marking M' ($=M{\cup}t\overset{N}{\bullet}{-}\overset{N}{\bullet}t$). This is denoted by $M[N, t\rangle M'$. A marking M' is said to be reachable from a marking M if there exists a firing sequence of transitions transforming M to M'. The set of all possible markings reachable from M is denoted by $R(N, M)$.

There are two important subclasses of WF-nets: *WS* and *EFC*. A structural characterization of good workflows is that two paths initiated by a transition/place should not be joined by a place/transition. WS is derived from this structural characterization. To give the formal definition of WS, we introduce some notations. The Petri net obtained by connecting p_O with p_I via an additional transition t^* is called the *short-circuited net* of N, denoted by $\overline{N}$ ($=(P, T{\cup}\{t^*\}, A{\cup}\{(p_O, t^*), (t^*, p_I)\}, \ell{\cup}\{(t^*, \tau)\})$). Let c be an elementary[**] circuit in $\overline{N}$. An elementary path ρ from a node n_1 ($\in P{\cup}T$) to another node n_2 in N is said to be a *handle* of

[**]A path/circuit is called elementary if no nodes appear more than once in the path/circuit.

c iff $\rho \cap c = \{n_1, n_2\}$. A handle from a transition to a place is called a TP-handle. A handle from a place to a transition is called a PT-handle. A WF-net N is WS if there are neither TP-handles nor PT-handles in $\overline{N}$. A WF-net is EFC if $\forall p_1, p_2 \in P$: $p_1\overset{N}{\bullet} \cap p_2\overset{N}{\bullet} \neq \emptyset \Rightarrow p_1\overset{N}{\bullet} = p_2\overset{N}{\bullet}$. Acyclic WS is a subclass of acyclic EFC.

There are two important properties in WF-nets: *soundness* and *branching bisimilarity*. Soundness is a criterion of logical correctness.

Definition 2 (soundness [1]): A WF-net N is sound iff (i) $\forall M \in R(N, [p_I])$: $\exists M' \in R(N, M)$: $M' \geq [p_O]$; (ii) $\forall M \in R(N, [p_I])$: $M \geq [p_O] \Rightarrow M = [p_O]$; and (iii) there is no dead transition in $(N, [p_I])$. □

Branching bisimilarity is widely used as an equivalence relation on WF-nets. Branching bisimilarity intuitively equates WF-nets which have the same external behavior. The behavior of a WF-net N $(=(P, T, A, \ell))$ is captured by the reachability graph of $(N, [p_I])$. It is denoted by $G=(V, E)$, where $V=R(N, [p_I])$, $E=\{(M, \ell(t), M') | M, M' \in V, t \in T, M[N, t\rangle M'\}$. Let $M, M' \in V$, $\alpha \in \ell(T)$. We write $M[N, \alpha\rangle M'$ if M' is reachable from M by following an edge labeled as α. We write $M[N, \tau^*\rangle M'$ if M' is reachable from M by following any number of edges labeled as τ. We write $M[N, (\alpha)\rangle M'$ if either (i) $\alpha = \tau$ and $M = M'$; or (ii) $M[N, \alpha\rangle M'$.

Definition 3 (branching bisimilarity [5]): Let G_X and G_Y be the reachability graphs of a WF-net $(N_X, [p_I^X])$ and another WF-net $(N_Y, [p_I^Y])$, respectively. A binary relation $\mathcal{R}$ $(\subseteq R(N_X, [p_I^X]) \times R(N_Y, [p_I^Y]))$ is branching bisimulation iff (i) if $M_X \mathcal{R} M_Y$ and $M_X[N_X, \alpha\rangle M'_X$, then $\exists M'_Y, M''_Y \in R(N_Y, [p_I^Y])$: $M_Y[N_Y, \tau^*\rangle M''_Y$, $M''_Y[N_Y, (\alpha)\rangle M'_Y$, $M_X \mathcal{R} M''_Y$, and $M'_X \mathcal{R} M'_Y$; (ii) if $M_X \mathcal{R} M_Y$ and $M_Y[N_Y, \alpha\rangle M'_Y$, then $\exists M'_X, M''_X \in R(N_X, [p_I^X])$: $M_X[N_X, \tau^*\rangle M''_X$, $M''_X[N_X, (\alpha)\rangle M'_X$, $M''_X \mathcal{R} M_Y$, $M'_X \mathcal{R} M'_Y$; and (iii) if $M_X \mathcal{R} M_Y$ then $(M_X = [p_O^X] \Rightarrow M_Y[N_Y, \tau^*\rangle [p_O^Y])$ and $(M_Y = [p_O^Y] \Rightarrow M_X[N_X, \tau^*\rangle [p_O^X])$. $(N_X, [p_I^X])$ and $(N_Y, [p_I^Y])$ are called *branching bisimilar*, denoted by $(N_X, [p_I^X]) \sim_b (N_Y, [p_I^Y])$, iff there exists a branching bisimulation $\mathcal{R}$ between G_X and G_Y. □

3. Acyclic EFC WF-Net Refactoring Problem and Its Properties

3.1 Definition and Analysis

We first give a formal definition of acyclic EFC WF-net refactoring problem.

Definition 4: Acyclic EFC WF-net refactoring problem
Input: Acyclic EFC WF-net $N_X = (P_X, T_X, A_X, \ell_X)$, where every external action is unique in N_X
Output: Acyclic WS WF-net $N_Y = (P_Y, T_Y, A_Y, \ell_Y)$
Constraints: (i) $(N_X, [p_I^X]) \sim_b (N_Y, [p_I^Y])$; (ii) every external action is unique in N_Y. □

Condition (ii) prohibits duplication of any external action. An external action is performed by resources (workers and/or machines). If the action is duplicated, it would share the resources with its duplicate. This makes it difficult that those resources are scheduled. Furthermore the action in the worst case is duplicated exponentially. This may disable refactoring from running in polynomial time.

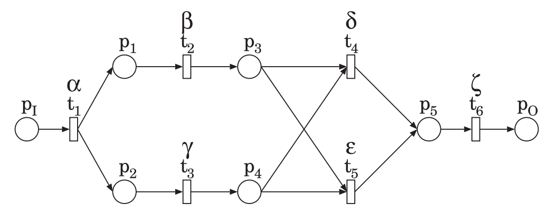

Fig. 1 An acyclic EFC but non-WS WF-net N_1.

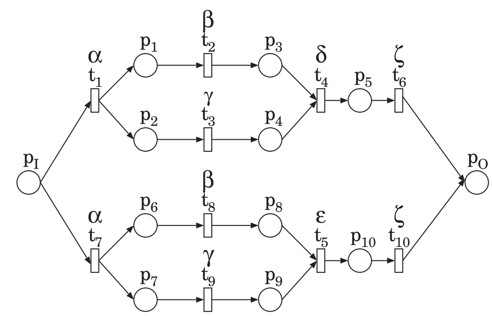

Fig. 2 An acyclic WS WF-net $N_1{}'$, which is obtained by using a trivial algorithm. $N_1{}'$ is not an answer of the refactoring problem, because external actions α, β, γ, and ζ are not unique.

There seems to exist a trivial algorithm for solving this problem. Let N be a sound EFC WF-net. It is known from Theorem 1 of Ref. [1] that $(\overline{N}, [p_I])$ is live and safe. From Theorem 14 of Ref. [2], we can view $\overline{N}$ as an interconnection of strongly connected MG-components†. All the strongly connected MG-components share t^* as an articulation point, where t^* is the transition connecting p_O and p_I in $\overline{N}$. Therefore for each of the strongly connected MG-components, we can obtain a MG WF-net†† by removing t^* from the MG-component. Connecting all the MG WF-nets so that those source places and those sink places are respectively shared, we can obtain an acyclic WS WF-net. Let us consider an application example of this trivial algorithm. Figure 1 shows an acyclic EFC but non-WS WF-net, denoted by N_1. Applying the trivial algorithm to N_1, we can obtain an acyclic WS WF-net, denoted by $N_1{}'$, which is shown in Fig. 2. Unfortunately, $N_1{}'$ is not an answer of the refactoring problem because it does not satisfy Constraint (ii), i.e. external ac-

†A MG-component N_1 of a Petri net N is defined as a subnet generated by transitions in N_1 having the following two properties: (i) Each place in N_1 has at most one incoming arc and at most one outgoing arc; and (ii) A subnet generated by transitions is the net consisting of these transitions, all of their input and output places, and their connecting arcs [2]. A Petri net is strongly connected if, for every pair of nodes n_1 and n_2, it contains a directed path from n_1 to n_2 and a directed path from n_2 to n_1.

††A WF-net is marked graph, MG for short, if $|p_I\overset{N}{\bullet}| = |\overset{N}{\bullet}p_O| = 1$ and $\forall p \in (P - \{p_I, p_O\})$: $|\overset{N}{\bullet}p| = |p\overset{N}{\bullet}| = 1$.

tions α, β, γ, and ζ are not unique. Therefore the trivial algorithm cannot solve the refactoring problem.

We give a decision problem related to the acyclic EFC WF-net refactoring problem.

Definition 5: EFC-WF-REFACTORING
Instance: Acyclic EFC WF-net $N_X = (P_X, T_X, A_X, \ell_X)$, where every external action is unique in N_X.
Question: Is there an acyclic WS WF-net $N_Y = (P_Y, T_Y, A_Y, \ell_Y)$ such that (i) $(N_X, [p_I^X]) \sim_b (N_Y, [p_I^Y])$; and (ii) every external action is unique in N_Y? □

3.2 Necessary Condition

We give a necessary condition for EFC-WF-REFACTORING.

Property 1: Let N_X be an acyclic EFC but non-WS WF-net whose every external action is unique. If N_X is not sound then there is no acyclic WS WF-net N_Y such that $(N_X, [p_I^X]) \sim_b (N_Y, [p_I^Y])$ and every external action is unique in N_Y. □

Proof: Since N_X is not sound, there is a dead marking besides $[p_O^X]$ in $(N_X, [p_I^X])$. The dead marking is denoted by M^X_{dead}. On the other hand, any acyclic WS WF-net is sound because its short-circuited net has neither TP-handles nor PT-handles. Since N_Y is sound, there is no dead marking besides $[p_O^Y]$ in $(N_Y, [p_I^Y])$. Therefore there is no branching bisimulation relation $\mathcal{R}$ such that $\exists M^Y \in R(N_Y, [p_I^Y]) : M^X_{dead} \mathcal{R} M^Y$. Thus this property holds. **Q.E.D.**

This property implies that we cannot solve the refactoring problem if a given acyclic EFC WF-net is not sound. It is known from Corollary 1 of Ref. [1] that an EFC WF-net can be checked for soundness in polynomial time. Thus the above necessary condition can also be checked in polynomial time.

3.3 Sufficient Condition

Let us consider the net shown in Fig. 1. In this net, well-structuredness is lost on account of the structure composed of p_3, p_4, t_4, t_5. We focus on such a structure, named *cross structure*.

Definition 6 (TP-cross and PT-cross structures): Let $N = (P, T, A, \ell)$ be an EFC WF-net.

- For places p_1, p_2 $(\in P)$ and transitions t_1, t_2 $(\in T)$, (t_1, t_2, p_1, p_2) is a TP-cross structure if $\{(t_1, p_1), (t_1, p_2), (t_2, p_1), (t_2, p_2)\} \subseteq A$.
- For places p_1, p_2 $(\in P)$ and transitions t_1, t_2 $(\in T)$, (p_1, p_2, t_1, t_2) is a PT-cross structure if $\{(p_1, t_1), (p_1, t_2), (p_2, t_1), (p_2, t_2)\} \subseteq A$. □

We define a subclass of acyclic EFC but non-WS, named *cross-bridged* (CB for short). A CB WF-net intuitively has only one cross structure, which is a cut-set of the net. To give the formal definition of CB, we use an operator: Given two WF-nets K and L, *place refinement* of a place p of K with L yields a WF-net $N = K \otimes_p L$, which is built as follows: p is replaced in K by L; transitions of $\overset{K}{\bullet}p$ become input transitions of the source place of L; and transitions of $p\overset{K}{\bullet}$ become output transitions of the sink place of L. The formal definition of CB is given as follows:

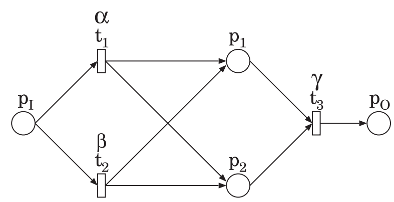

Fig. 3 A CB WF-net, N_{TP}^{CB}, with one TP-cross structure.

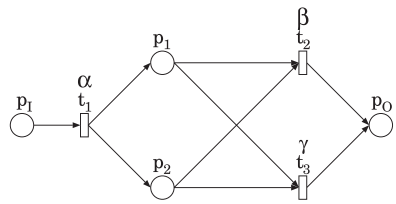

Fig. 4 A CB WF-net, N_{PT}^{CB}, with one PT-cross structure.

Definition 7 (cross-bridged; CB): A WF-net N is cross-bridged, CB for short, if N is

- $(((N_{TP}^{CB} \otimes_{p_I} L_I) \otimes_{p_1} L_1) \otimes_{p_2} L_2) \otimes_{p_O} L_O$, where N_{TP}^{CB} is the net shown in Fig. 3, $\{p_I, p_1, p_2, p_O\}$ is the set of places in N_{TP}^{CB}, and L_I, L_1, L_2, L_O are acyclic WS WF-nets.
- $(((N_{PT}^{CB} \otimes_{p_I} L_I) \otimes_{p_1} L_1) \otimes_{p_2} L_2) \otimes_{p_O} L_O$, where N_{PT}^{CB} is the net shown in Fig. 4, $\{p_I, p_1, p_2, p_O\}$ is the set of places in N_{PT}^{CB}, and L_I, L_1, L_2, L_O are acyclic WS WF-nets. □

Now we give a sufficient condition for EFC-WF-REFACTORING.

Property 2: Let N_X be an acyclic EFC but non-WS WF-net whose every external action is unique. If N_X is CB then there is an acyclic WS WF-net N_Y such that $(N_X, [p_I^X]) \sim_b (N_Y, [p_I^Y])$ and every external action is unique in N_Y. □

Proof: This proof consists of the following two cases: (i) N_X is $(((N_{TP}^{CB} \otimes_{p_I} L_I) \otimes_{p_1} L_1) \otimes_{p_2} L_2) \otimes_{p_O} L_O$, where N_{TP}^{CB} is the net shown in Fig. 3, $\{p_I, p_1, p_2, p_O\}$ is the set of places in N_{TP}^{CB}, and L_I, L_1, L_2, L_O are acyclic WS WF-nets. (ii) N_X is $(((N_{PT}^{CB} \otimes_{p_I} L_I) \otimes_{p_1} L_1) \otimes_{p_2} L_2) \otimes_{p_O} L_O$, where N_{PT}^{CB} is the net shown in Fig. 4, $\{p_I, p_1, p_2, p_O\}$ is the set of places in N_{PT}^{CB}, and L_I, L_1, L_2, L_O are acyclic WS WF-nets.

Case (i): Let us first consider a special case, i.e. N_X is N_{TP}^{CB}. Assume that N_Y is the WF-net N_{TP}^{WS} shown in Fig. 5. N_{TP}^{WS} is obviously acyclic WS. We must that N_{TP}^{WS} satisfies Constraints (i) and (ii) of the acyclic EFC WF-net refactoring problem. In order to show that N_{TP}^{WS} satisfies Constraint (i), we assume the following relation $\mathcal{R}$ between $R(N_{TP}^{CB}, [p_I])$ and $R(N_{TP}^{WS}, [p_I])$: $[p_I] \mathcal{R} [p_I]$, $[p_1, p_2] \mathcal{R} [p_3]$, $[p_1, p_2] \mathcal{R} [p_1, p_2]$, $[p_O] \mathcal{R} [p_O]$. We show that $\mathcal{R}$ satisfies Condition (i) of branching bisimilarity. If $[p_I] \mathcal{R} [p_I]$ and $[p_I][N_{TP}^{CB}, \alpha\rangle[p_1, p_2]$

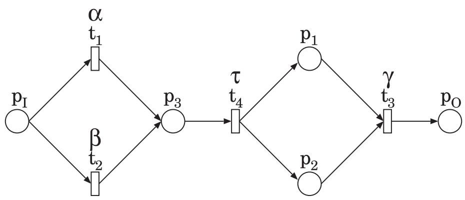

Fig. 5 An acyclic WS WF-net N_{TP}^{WS}.

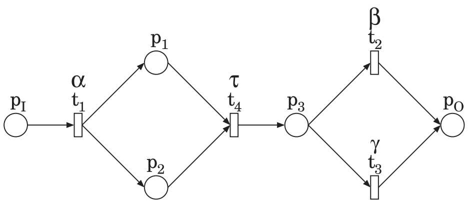

Fig. 6 An acyclic WS WF-net N_{PT}^{WS}.

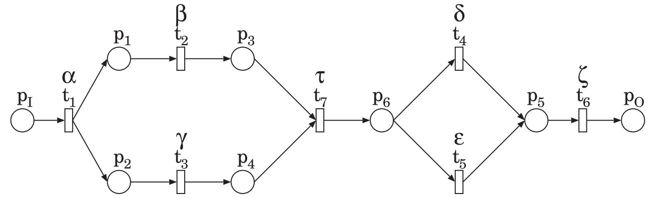

Fig. 7 An answer of the acyclic EFC WF-net refactoring problem for N_1.

then $[p_I][N_{TP}^{WS},\alpha\rangle[p_3]$ and $[p_1,p_2]\mathfrak{R}[p_3]$ hold. If $[p_I]\mathfrak{R}[p_I]$ and $[p_I][N_{TP}^{CB},\beta\rangle[p_1,p_2]$ then $[p_I][N_{TP}^{WS},\beta\rangle[p_3]$ and $[p_1,p_2]\mathfrak{R}[p_3]$ hold. If $[p_1,p_2]\mathfrak{R}[p_3]$ and $[p_1,p_2][N_{TP}^{CB},\gamma\rangle[p_O]$ then $[p_3][N_{TP}^{WS},\tau\rangle[p_1,p_2]$, $[p_1,p_2]\mathfrak{R}[p_1,p_2]$, $[p_1,p_2][N_{TP}^{WS},\gamma\rangle[p_O]$, and $[p_O]\mathfrak{R}[p_O]$ hold. We show that $\mathfrak{R}$ satisfies Condition (ii) of branching bisimilarity. If $[p_I]\mathfrak{R}[p_I]$ and $[p_I][N_{TP}^{WS},\alpha\rangle[p_3]$ then $[p_I][N_{TP}^{CB},\alpha\rangle[p_1,p_2]$ and $[p_1,p_2]\mathfrak{R}[p_3]$ hold. If $[p_I]\mathfrak{R}[p_I]$ and $[p_I][N_{TP}^{WS},\beta\rangle[p_3]$ then $[p_I][N_{TP}^{CB},\beta\rangle[p_1,p_2]$ and $[p_1,p_2]\mathfrak{R}[p_3]$ hold. If $[p_1,p_2]\mathfrak{R}[p_3]$ and $[p_3][N_{TP}^{WS},\tau\rangle[p_1,p_2]$ then $[p_1,p_2]\mathfrak{R}[p_1,p_2]$ holds. If $[p_1,p_2]\mathfrak{R}[p_1,p_2]$ and $[p_1,p_2][N_{TP}^{WS},\gamma\rangle[p_O]$ then $[p_1,p_2][N_{TP}^{CB},\gamma\rangle[p_O]$ and $[p_O]\mathfrak{R}[p_O]$ hold. $\mathfrak{R}$ obviously satisfies Condition (iii) of branching bisimilarity. Therefore $(N_{TP}^{CB},[p_I]) \sim_b (N_{TP}^{WS},[p_I])$ holds. And we can know from the structure of N_{TP}^{WS} that N_{TP}^{WS} satisfies Constraint (ii) of the refactoring problem, i.e. every external action is unique. Thus if N_X is N_{TP}^{CB} then this property holds.

Next let us consider a general case, i.e. N_X is $(((N_{TP}^{CB}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O$. Assume that N_Y is $(((N_{TP}^{WS}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O$. The net is acyclic WS because place refinement of a place in an acyclic WS WF-net with an acyclic WS WF-net yields an acyclic WS WF-net. Since $(N_{TP}^{CB},[p_I]) \sim_b (N_{TP}^{WS},[p_I])$ and L_I, L_1, L_2, L_O are acyclic WS WF-nets, $((((N_{TP}^{CB}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O, [p_I]) \sim_b ((((N_{TP}^{WS}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O,[p_I])$ holds. And we can know from the net structure that every external action is unique. Thus if N_X is $(((N_{TP}^{CB}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O$ then this property holds.

Case (ii): Let us first consider a special case, i.e. N_X is N_{PT}^{CB}. Assume that N_Y is the WF-net N_{PT}^{WS} shown in Fig. 6. N_{PT}^{WS} is obviously acyclic WS. Assume the following relation $\mathfrak{R}$ between $R(N_{PT}^{CB},[p_I])$ and $R(N_{PT}^{WS},[p_I])$: $[p_I]\mathfrak{R}[p_I]$, $[p_1,p_2]\mathfrak{R}[p_1,p_2]$, $[p_1,p_2]\mathfrak{R}[p_3]$, $[p_O]\mathfrak{R}[p_O]$. We can prove $(N_{PT}^{CB},[p_I]) \sim_b (N_{PT}^{WS},[p_I])$ in a similar way as Case (i). And we can know from the net structure that every external action is unique. Thus if N_X is N_{PT}^{CB} then this property holds.

Next let us consider a general case, i.e. N_X is $(((N_{PT}^{CB}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O$. Assume that N_Y is $(((N_{PT}^{WS}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O$. We can prove $((((N_{PT}^{CB}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O,[p_I]) \sim_b ((((N_{PT}^{WS}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O,[p_I])$ holds in a similar way as Case (i). And we can know from the net structure that every external action is unique. Thus if N_X is $(((N_{PT}^{CB}\otimes_{p_I}L_I)\otimes_{p_1}L_1)\otimes_{p_2}L_2)\otimes_{p_O}L_O$ then this property holds.

Q.E.D.

This property implies that we can solve the refactoring problem if a given acyclic EFC WF-net is CB.

Let us consider the computation complexity of checking the sufficient condition. We give an algorithm for checking whether a given acyclic EFC but non-WS WF-net is CB.

$\ll$ Decision of Cross-Brigdedness $\gg$

Input: Acyclic EFC but non-WS WF-net N $(=(P,T,A,\ell))$

Output: Is N CB?

1° Find cross structures in N. If two or more cross structures are found, output no and stop.

2° For each pair $(p_1,p_2) \in P\times P$, if any path from p_I to p_2 includes p_1, and any path from p_1 to p_O includes p_2, then let N' be a WF-net obtained by connecting all paths from p_1 to p_2, apply $\ll$ Decision of Well-Structuredness $\gg$ of Ref. [3] to N'. If the result is yes, reduce the part corresponding to N' in N to a place.

3° If the resultant net is isomorphic with N_{TP}^{WS} or N_{PT}^{WS} then output yes and stop. Otherwise output no and stop.

Since $\ll$ Decision of Well-Structuredness $\gg$ is a polynomial time algorithm, $\ll$ Decision of Cross-Brigdedness $\gg$ can run in polynomial time obviously. Thus the above sufficient condition can be checked in polynomial time.

Let us consider the net N_1 shown in Fig. 1. We can transform N_1 to a net which is isomorphic to N_{PT}^{CB}. This implies that N_1 is CB. We can know from the sufficient condition that we can solve the refactoring problem for N_1. Figure 7 shows an answer of the problem.

4. Conclusion

In this paper, we have first given the formal definition of acyclic EFC WF-net refactoring problem. Next we have given a necessary condition and a sufficient condition for solving the problem. Then we have shown that those conditions can be checked in polynomial time. Finally we have illustrated that an instance of the problem can be solved with those conditions. Those conditions are only the first step to the refactoring problem. As the next step, we plan to investigate refactoring for the more general case. To promote this plan, we first investigate decidability of the decision problem, EFC-WF-REFACTORING, related to the refactoring problem. If the problem is decidable then we would give a refactoring method for refactorable acyclic EFC WF-nets. Otherwise, we would look for a larger subclass of acyclic EFC refactorable to acyclic WS, and then give a refactoring method for the subclass. The method is intuitively to remove structures not allowed in acyclic WS WF-nets from a given EFC WF-net, guaranteeing branching bisimilarity.

Acknowledgements

The author would like to thank Mr. Yuki Kuroda for his contribution to this research. This research was supported by KAKENHI 23560532.

References

[1] W.M.P. van der Aalst, "The application of Petri nets to workflow management," J. Circuits Syst. Comput., vol.8, no.1, pp.21–65, 1998.

[2] T. Murata, "Petri nets: Properties, analysis and applications," Proc. IEEE, vol.77, no.4, pp.541–580, 1989.

[3] N. Qu, S. Yamaguchi, and Q.W. Ge, "Parallel degree of well-structured workflow nets," IEICE Trans. Fundamentals, vol.E93-A, no.12, pp.2730–2739, Dec. 2010.

[4] M. Fowler, Refactoring: Improving the Design of Existing Code, Addison Wesley, 1999.

[5] H.M.W. Verbeek and T. Basten, "Deciding life-cycle inheritance on Petri nets," Proc. ICATPN 2003, pp.44–63, 2003.