Applied Mathematics & Information Sciences
An International Journal

© 2012 NSP
Natural Sciences Publishing Cor.

The Analysis Method about Change Domain of Business Process Model Based on the Behavior Profile of Petri Net

Xianwen Fang[1], *WenJun Hao*[1], *Xiaoqin Fan*[2] *and Zhixiang Yin*[1]

[1] Department of Information and Computing Science, Anhui University of Science and Technology,Huainan 232001, China
[2] School of Information and Computer Science, Shanxi University,Taiyuan 030051, China

Received: Oct 8, 2011; Revised Apr. 20, 2012; Accepted May. 4, 2012

Abstract: For adapting flexibly the rapidly changing of business needs, the analysis of the business process change domain (or change nodes) to become one of the core of business process management. Existing research methods are based in the static analysis that searching the change region by given nodes, which had big limitations to the uncertain task nodes. In order to determine the changes domain of business process models for the uncertain task node, a dynamic analysis method about the change domain of business process is proposed based on the behavior profile of the Petri net in the paper. The method uses the behavior profile to analyze the behavior order relationships of business process model, look for the change domain and suspicious node set in the process model, and to further determine the change node set in the change domain. Finally, for analyzing the effectiveness of the method, a specific change domain analyzing instance of process model is given out. Theory and case study shows that the proposed analysis methods has big advantage to the change domain analyzing of the process model under existing uncertain task node circumstance.

Keywords: Process model, Petri net, Behavior profile, Change domain, Change node.

1. Introduction

With the development of computer technology, business process applications continues to expand more in the theoretical and technical aspects of the business process modeling to guide the design to meet the special requirements of process models. At present, it is an important task to translate business requirements into system specifications of the software design engineering in the business environment. However, due to the different expectation goals and the abstract level of business analysis and systems analysis that companies design different target models from the same source model, its important to discuss the consistency between these corresponding models.

Checking the consistency between process models is the core issue of the theory and practice of process modeling, we looking for inconsistencies range between these models to determine the change region in the process model which is also a key issue to solve the problem that does not match each other and optimize the model gradually.

Now, there are many scholars in the study of the consistency between process models [1–4]. Matthias Weidlich compared the behavior profile and trace equivalence [1] when research the consistency measurement of process models, and showed behavior profile can be computed more efficiently which its advantage on the measurement. Introduce two major categories of noise that leads to noncompliance of process models based on behavior profile are discussed in [2], the literature [3], was about process compliance measurement based on behavior profiles. Genetic algorithms [4] were used to find change region based on behavior aware, considering the matching between the target model and the copy edition of source model. Weidlich [5] researched how to narrow the region when change propagation in process models. But the authors did not discuss how to find the change node, they assumed that a change can be localized by a dedicated node; the purpose is to explore the propagation of changes between related process models. These are static considerations based on behavior profile that captures constraint relation between

---

* Corresponding author: e-mail: fangxianwen@hotmail.com; zxyin66@163.com.

activity nodes; it didn't make much analysis of the dynamic changes in behavior cased by the activities.

In this paper, our goal is to build a process model based on Petri nets, giving its definition and related concepts, combining with Petri net dynamic behavior characteristics and behavior profile that captures control relations between each pair of activities from corresponding models. First, we can find the suspicious area in source model, and then construct some algorithms to searching and narrowing the area. Finally, we determine the change range and get the change node.

The reminder of this article is structured as follows. Section 2 introduces the basic notations. According the relate knowledge of Petri nets and behavior profile, giving two important algorithms to determine the change node in Section 3. We report our findings of applying these algorithms to the reference model in section 4. Finally, section 5 concludes the paper and provides an outline on the future work.

2. Basic conception

This part introduces some basic concepts used in this article, and other related concepts can be found in [1,5,6]. First, we introduce the nation of a process Petri net used throughout the paper. Second, we define behavior profiles to capture the behavior characteristics of the process network. Finally, introduces the correspondence between the process models network.

Definition 1*(Process Petri Net) A process Petri net is a tuple* $BP_p = (P, T, F, C)$ *with*

- P *is a set of condition nodes,* T *is a set of change activity nodes, such that* $P \neq \emptyset$*,* $T \neq \emptyset$ *and* $P \cap T = \emptyset$*;*
- $F \subseteq (P \times T) \cup (T \times P)$ *is the flow relation;*
- $C = \{od, or, pl, cy\}$ *as the type of process network structure, ie order, or, parallel and cycle structures.*

A process Petri net can be made of those four structures from a business process. Given an initial marking of a Petri net, it exists a firing sequence because of an occurrence of rule. There are some related concepts about Petri net, like the set of all input and output places of a transition, respectively, markings, and firing sequences and reachable and so on can refer to [6]. The notion of observed execution sequences of process Petri net is formalized as follows:

Definition 2*(Execution Sequence) the execution sequence* T_{pm} *for a process Petri net* $BP_p = (P, T, F, C)$*, is the set of all firing sequences* $\sigma = t_1 t_2 \ldots t_{n-1} t_n > 0$*,* $n \in N^*$ *for which holds* $M_0[\sigma \rightarrow M_n$*.*

The above definition explains if a process model Petri net $BP_p = (P, T, F, C)$ exists an observed execution sequence, there is a path from start activity node to the end one, to ensure that the process model network is live, not deadlock or dead transition.

Behavior profile captures the behavior relation between processes in graze-fine. The net proposed in this paper contains four types of relations; they are all established on the basis of feasible trace. Two nodes or flow relation of a process model net are in the same feasible trace, if there exists a sequence in which one node occurs after the other.

Definition 3*(Feasible Trace) A process Petri net* $BP_p = (P, T, F, C)$*, execution sequence* T_{pm}*,* $\tau = n_1 n_2 \ldots n_k$ *is a set of nodes, if* $(x, y) \subseteq ((N \cup F)(N \cup F))$*, such that* $j \in (1, \ldots, k-1)$*,* $j < h \leq k$*, which holds* $n_j = x$*,* $n_h = y$*, so* τ *a feasible trace, and* $\tau \in T_{pm}$*, holds* $x \prec y$*.*

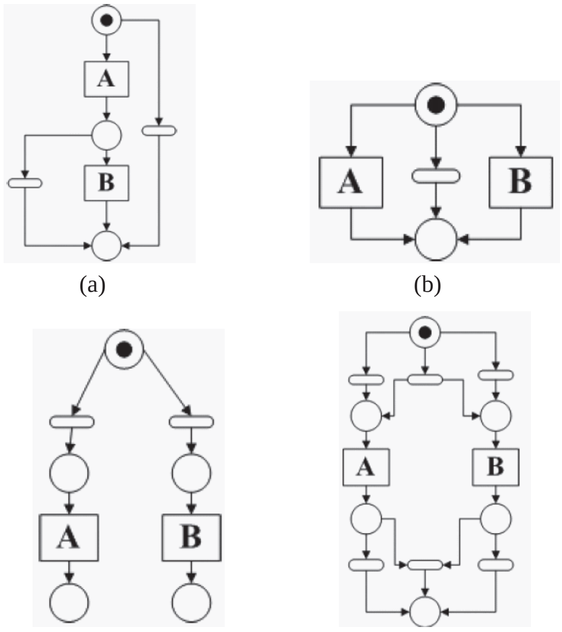
The behavior profile is grounded on the notion of feasible trace between activity transition nodes in a process Petri net. This relation requires the existence of such a trace and does not have to hold for all traces of the model. Depending on how two activity transition nodes of a process Petri net are related by feasible trace, we define four relations forming the behavior profile.

Definition 4*[2], (Behavior Profile) Let* $BP_p = (P, T, F, C)$ *be a process Petri net,* x*,* y *are activity transition nodes,* $(x, y) \subseteq ((N \cup F)(N \cup F))$ *is in one of the following relation:*

- *The strict order relation* $\rightarrow (x, y)$*, if* $x \prec y$*, and* $y \not\prec x$*;*
- *The exclusiveness relation* $+(x, y)$*, if* $x \not\prec y$*,* $y \not\prec x$*;*
- *The parallel relation* $||(x, y)$*, if* $x \prec y$*,* $x \prec z$*;*
- *The interleaving order relation* $\times(x, y)$*, if* $x \prec y$*,* $y \prec x$*.*

The set of all four relations is the behavior profile of BP_p. Note that we say that a pair (x, y) is in reverse order relation, denoted by $\rightarrow^{-1} (x, y)$, if $y \prec x$ and $x \not\prec y$.

A process Petri net $BP_p = (P, T, F, C)$, M_0 is an initial marking, we can determine the relationship between activity transition nodes in process model. The exemplary process in Figure 1 illustrates the relations of the behavior profile for two dedicated activity transition nodes A and B. The Figure 1(a) shows that the “strict” order relation enforces both nodes occur together or neither of them occurs. In the Figure 1(b), “and” order relation captures the fact that all occurrences of both activity transition nodes are ordered in all firing sequences reachable from the initial marking, i.e. the occurrence of A lead to the occurrence of B, denoted as $A \rightarrow B$. In the Figure 1(c), both activity transitions are exclusive to each other, $A+B$. That is, both nodes are never observed together in any firing sequence, or there may be a firing sequence that contain none of them. In the Figure 1(d) shows the parallel order relation between the two activity condition nodes that can be observed in two branch line running in parallel rode in a firing sequence under the same initial marking denoted $A||B$. The last one exemplify interleaving order of two activity transition nodes, $A \times B$. It does not necessarily imply the concurrent enabling of two nodes in a certain marking, but the two nodes can exist in any order in a firing sequence. It might also result from cyclic structure.



(a) (b)

(c) (d)

Figure 1 The structure relation of behavior profile

In order to find the change region in the target model, first and foremost, we have to clarify the notion of the corresponding relation between process models and propose the correspondences between activity transition nodes in process Petri nets based on behavior profile.

Definition 5*(Correspondence Relation based on Transition Pairs). Two process Petri nets $BP_{p1} = (P_1, T_1, F_1, C_1)$ and $BP_{p2} = (P_2, T_2, F_2, C_2)$, Correspondence relation $\sim \subseteq T_1 \times T_2$, if there is a mapping when $\forall (t_a, t_s), (t_b, t_y) \in \sim$ and $\forall t_1 \in T_1$, $\exists t_2 \in T_2$ which holds $t_1 \sim t_2$, then $(t_a = t_b) \Rightarrow (t_x = t_y)$, include all activity transition nodes in process Petri net.*

In order to find the pairs of activity transition nodes that satisfy the correspondence relation between source model and target model, according the dynamic thinking of Petri net, given the algorithm to derive the nodes that have the correspondence relation. The algorithm is showed in algorithm 1.

For a business process, a change operation in target model is reflected in its behavior profile, i.e., its characteristic relations. Therefore, these relations localize the change in the target model. Under the assumption that the target model and the source model have been aligned to some extent, we can localize the change region in target model by the corresponding activities.

Definition 6*[5], (Change Region) A process Petri net $BP_p = (P, T, F, C)$, a change region $BP_p' = (P', T', F', C')$ is a subset of BP_p which holds $P' \in P, T' \in T, F' \in F, C' \in C$ and $BP_p' \subseteq BP_p$.*

Algorithm 1 Looking for the set of corresponding transitions

```
Input:  $BP_{p1} = (P_1, T_1, F_1, C_1)$ process Petri net 1
        $BP_{p2} = (P_2, T_2, F_2, C_2)$ process Petri net 2.
Output: $A$, the set of corresponding transitions.
1: According to the definition 5 obtain a suspicious area $W_{01}$, marked the nodes $a_{01}a_{02}a_{03} \wedge a_{0n-1}a_{0n}$, the area of corresponding source model is $W_{01}'$.
2: From $a_{01}$ to consider its behavior relationship with the next node.
3: Compare the behavior relationship in source model, if $a_{01} \rightarrow a_{02}$, the suspicious area is $W_{01} - \{a_{01}\}$.
4: When meet branching structure, if $+(a_{0i-1}, a_{0i})$, $2 \leq i \leq n$, choose one branching compare to the correspondent structure of source model, if not satisfy the order relation, $a_{0i-1}$ is a suspicious node, else suspicious area is $W_{01} - \{a_{01}, a_{0n}, a_{0i}\}$; if $||(a_{0i-1}, a_{0i})$ or $\times(a_{0i-1}, a_{0i})$, $2 \leq i \leq n$, return to step 2.
5: Get a change region $W_{01} - \{a_{01}, a_{0n}, a_{0i}, \ldots\}$, $2 \leq i \leq n$, a set of suspicious node $Q$. Output change region $W_0$ and suspicious nodes $Q$.
```

Initially, the change region is a subnet of the target model, and it is narrowed using the correspondence between target model and source model.

3. The analysis method of looking for change domain based on the dynamic behavior of Petri Net

A change operation in the source business process model is reflected in its behavior profile, i.e., its characteristic relations. So these relations locate the change in the source process model. Because the source and the target process model have been aligned to some extent, in order to locate a change region in the target process model, using the characteristic relations for the corresponding activities in the target process model. This region identifies a part of the target model in which the change of the source process model has to be realized [6,7].

Weidlich et. al [5,8] determined the change region might either be a set of flow arcs of the target model or an empty set. The former indicates that there are already flow arcs in the model that fulfil the requirements of the behavior profile with respect to the potential change. Therefore, all these flow arcs qualify for representing the change of the source model in the target model. Note that these flow arcs are not necessarily part of a connected subgraph of the process model. Instead, there might be multiple areas in the target model that qualify for the realization of the change. In case the set is empty, the target model does not yet contain a flow arc satisfying the behavioral requirements. However, it is always possible to insert flow arcs according behavioral requirements. In this case, the boundary nodes and inter-boundary nodes guide the adaptation of the target process model. The former impose requirements regarding

© 2012 NSP
Natural Sciences Publishing Cor.

the strict order relation, whereas the exclusiveness and observation concurrency relation have to be considered for the latter.

Some proposed methods mainly adopted the static behavior methods, the suspicious area [9,10] is analyzed by the appointed nodes, but this is a particular process.In order to localize the change region in a business process, we propose a dynamic behavior analysis method. First, according to definition 1,we transform the business into process Petri net. Second, determine the suspicious area based on the correspondence between source model and target model. Third, narrow the suspicious area gradually by the constraints of behavior profile. Finally, extract all the execution sequences that through the suspicious area, we use behavior relations [11] and the dynamic behavior characteristics of Petri net to find the smallest change region and change nodes [12–14] . We give Algorithm 2 to derive the optimal change region. Based on Algorithms 2, we propose Algorithm 3 to find the set of change nodes using the characteristics of the dynamic behavior of Petri net.

Algorithm 2 Determining the change region in target model

```
Input:  $BP_{p0} = (F_0, T_0, F_0, C_0)$, source model
        $BP_p = (P, T, F, C)$, target model
Output: $W_0$, change region
1: Get a suspicious area $W_{01}$ from Algorithm 1, marked the nodes $a_{01}a_{02}a_{03} \wedge a_{0n-1}a_{0n}$, the area of corresponding source model is $W_{01}^{'}$;
2: From $a_{01}$ to consider its behavior relationship with the next node;
3: Compare the behavior relationship in source model, if $a_{01} \rightarrow a_{02}$. suspicious area is $W_{01} - \{a_{01}\}$; else
4: From $a_{0n}$ to consider its behavior relationship with the last node st the same time;
5: Compare the behavior relationship in source model, if $a_{0n} \rightarrow^{-1} a_{0n-1}$, suspicious area is $W_{01} - \{a_{01}, a_{0n}\}$; else, continue
6: When meet branching structure, if $+(a_{0,i-1}, a_{0i})$, $2 \leq i \leq n$, choose one branching compare to the correspondent structure of source model, if not satisfy the order relation, $a_{0i-1}$ is a suspicious node ,else suspicious area is $W_{01} - \{a_{01}, a_{0n}, a0i\}$, turn to step 2;
7: If $||(a_{0i-1}, a_{0i})$ or $\times(a_{0i-1}, a_{0i})$, $2 \leq i \leq n$, turn to step 2;
8: Get a change region $W_{01} - \{a_{01}, a_{0n}, a_{0i}, \ldots\}$, $2 \leq i \leq n$, a set of suspicious node $Q$. Output change region $W_0$ and suspicious nodes' $Q$.
```

Algorithm 3 Looking for the set of change node

```
Input:  $W_0$, change region
        $Q$, a set of suspicious node
        $S = (N, M_0)$, a initial marking
Output: $N_0$, a set of change node
1: According to definition 1 transform the change region $W_0$ in target model into process petri net;
2: Given a initial marking $S = (N, M_0)$ to target model $BP_p = (P, T, F, C)$, get all firing sequences $\sigma_1\sigma_2 \wedge \sigma_m$ that through the suspicious node and change region, $W_c = W_{01} - \{a_{01}, a_{0n}, a_{0i}, \ldots\}, 2 \leq i \leq n$;
3: Select a sequence $\sigma_i$, $1 \leq i \leq m$, not repeat choose two different condition place nodes $p_i, p_j, i < j$;
4: If ${}^\bullet p_i \neq \emptyset$ or ${}^\bullet p_i \neq \emptyset$, $S_1 = p_i \cup {}^\bullet p_i \cup p_j \cup {}^\bullet p_i \ldots$, i.e. the set of all input transition of $p_i$ and $p_j$, and the set of all input places of transitions and so on; else $S_1 = \sigma_i$:
5: If ${}^\bullet p_i \neq \emptyset$ or ${}^\bullet p_i \neq \emptyset$, $S_2 = p_i \cup p_i{}^\bullet \cup p_j \cup p_j{}^\bullet \cup \ldots$, i.e. the set of output transition of $p_i$ and $p_j$ ,and the set of all output transitions of places and so on; else $S_1 = \sigma_i$.
6: Return to step 3.
```

4. Case study

In this paper, we select source model and target model in [5] as our examples. First, the Petri net of source model and target model are shown in Figure 2 and Figure 3, According to Algorithm 1, the dashed box part in Figure 3 is the initial suspicious area from the correspondence between models , Further on, we consider the constraint relations of activity transitions between models based on behavior profile , using Algorithm 2 to narrow the dashed box part until we find the smallest change region in target model which shown in Figure 4. According to Algorithm 3, we collect all observed execution sequences that through W_0 in Figure 4. There are three execution sequences fragment, i.e. $t_7t_9t_{15}$, $t_8t_{13}t_{15}$ and $t_8t_{11}t_{12}$. Take $t_8t_{13}t_{15}$ for example, given two condition places p_{11} and p_{13}, The first step, for the place p_{11}, Algorithm 3 returns the activity transitions T in ${}^\bullet p_{11}$ and the input places of these transitions ${}^\bullet T$ until $\exists n_i \notin W_{min}$, so does p_{13}. Then, move backwards adding the places and transitions become a sub-modules S_1 which shown in left side of Figure 5. The second step, for the place p_{11}, Algorithm 3 moves forwards adding the transitions $T^{'}$ in $p_{11}^{\bullet}$ and the output places of these transitions $T^{'\bullet}$ until $\exists n_j \notin W_{min}$, so does p_{13}. Then we get a sub-modules S_2 which shown in right side of Figure 5. The third step, the change node under the sequence $t_8t_{13}t_{15}$ is the intersection of S_1 and S_2, i.e. $S_1 \cap S_2 = \{t_{13}\}$. According the same analysis in the above method, we can also choose p_8 and p_{13}, the intersection is $\{t_9\}$, Any other group of places are the empty set. Finally, the change region in the target model is $\{t_9p_{13}t_{13}\}$, and the set of change nodes is $\{t_9, t_{13}\}$.

5. Conclusion

In order to adapt to rapidly changing business needs, looking for change nodes in business processes has a certain practical significance. In [5],they illustrate a change region can be localized by a dedicated node which is determined from the static respect. In this paper, we give a structure of business process model based on Petri net and propose

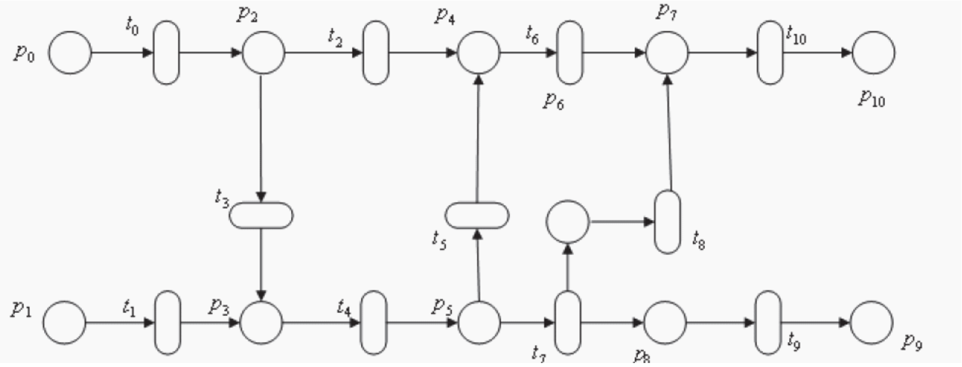


Figure 2 Petri net of source business process model

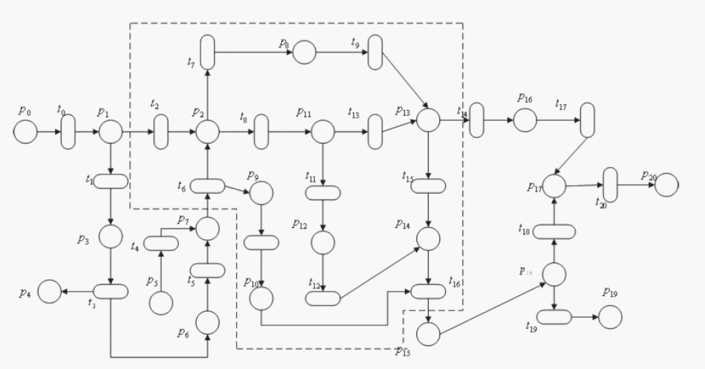


Figure 3 Petri net of target business process model

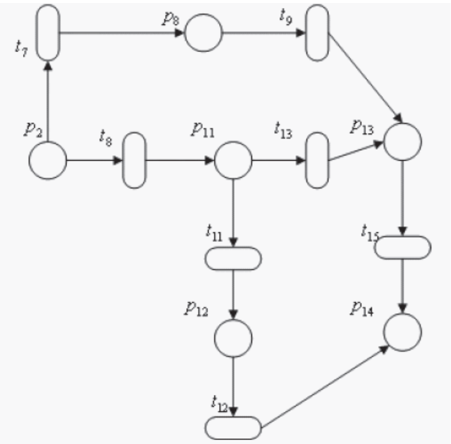
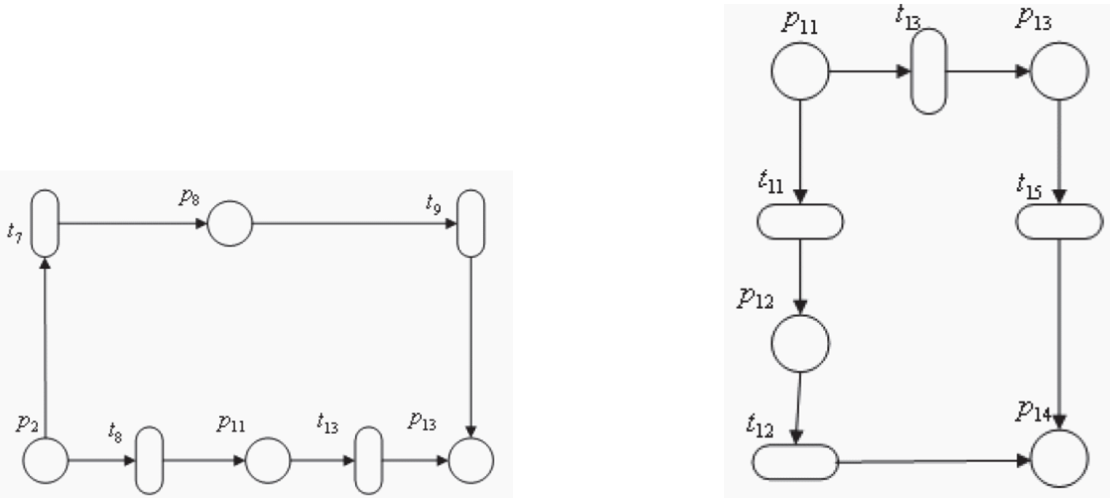


Figure 4 The smallest change region

© 2012 NSP



(a) (b)

Figure 5 Backward and forward set of p_{11} and p_{13}

the behavior profile of process Petri net. First of all, we give Algorithm 1 to discuss the correspondence between activity place nodes of source model and target model. Then combine the behavior profile and the dynamic behavior characteristics of Petri net to looking for a change region in Algorithm 2. Finally, Algorithm 3 is proposed to narrow the change region gradually and find a set of change node which based on Algorithm 2. Our approach overcomes the shortcomings that a process model can only be determined in a static node. And give a formal analysis of the dynamic behavior of the process model. Instance analysis shows that our method has certain validity.

As a future work, we plan to use our approach to check the consistency between the process models and carry on the amendment of the model with the change node. We can also optimize a target model gradually to make it close to its source model. And we hope that these operations can be realized automated processing in the near future.

Acknowledgement

We would like to thank the support of the National Natural Science Foundation of China under Grant No.61170059, No.61170172, No.61100058, No.60973050, No.61073102 and No.11126029, the National High-Tech Research and Development Plan of China under Grant No.2009AA01Z4-01, the Natural Science Foundation of Educational Government of Anhui Province of China (KJ2011A086 and KJ2012A073), Anhui Provincial Natural Science Foundation(1208085MF105), Anhui Provincial Soft Science Foundation(12020503031), the Natural Science Foundation of Shanxi (Nos. 2011021013-2).

References

[1] M. Weidlich, J. Mendling and M. Weske, IEEE Transactions Softare Engineering 37, 410 (2011).
[2] M.Weidlich, A. Polyvyanyy, N. Desai, J. Mendling and M. Weske, Information Systems 36, 1009 (2011).
[3] M. Weidlich and J. Mendling, Information Systems 37, 80(2012).
[4] M. Weidlich, A. Polyvyanyy, N. Desai and J. Mendling, Lecture Notes in Computer Science 6051, 499 (2010).
[5] M. Weidlich, M. Weske and J. Mendling, Proc of IEEE International Conference on Services Computing, 33(2009).
[6] T. Murata, Proc. IEEE. 11, 541 (1989).
[7] R. Dijkman, M. Dumas and C. Ouyang, Information and Software Technology 51, 1281 (2009).
[8] R. M. Dijkman, Lecture Notes in Computer Science 5240, 261 (2008).
[9] E Kindler, Lecture Notes in Computer Science 3080, 82 (2004).
[10] B. Weber, S. Rinderle and M. Reichert, Lecture Notes in Computer Science 4495, 574 (2007).
[11] M. Lorens, J. Oliver, J. Silva, S. Tamarit and G. Vidal, Electronic Notes in Theoretical Computer Science 233, 153 (2008).
[12] M. Weidlich, F. Elliger and M.Weske, Lecture Notes in Computer Science 6551, 101 (2011).
[13] X.W.Fang, C.J. Jiang, Z.X.Yin and X.Q.Fan, Computer Science and Information Systems 8, 843 (2011).
[14] K. Johannes, K. Markus and S. Tarja, Software—Practice & Experience 40, 701 (2010).

Xianwen Fang received his M.A. degree from Shandong University of Science and Technology, China, in 2004, and PhD. degree in the key Lab of Service Computing at Tongji University in 2011. He is currently a Professor with the Department of Computer Science and Engineering, Anhui University of Science and Technology, China. His research interests include Petri net, trustworthy software and Web services. He has published more than 60 papers in domestic and international academic journals and conference proceedings. These papers are embodied more than 40 times by SCI and EI and are cited more than 80 times by others.

Wenjun Hao received the M.A. degree from Anhui University of Science and Technology, China, in 2011. She is currently a lecturer with the Department of Computer Science and Engineering, Anhui University of Science and Technology, China. Her current areas of research are concurrent theory, Petri net and formal verification of software.

Xiaoqin Fan received her M.A. degree from TaiYuan University of Science and Technology, China, in 2002, and PhD. degree in the key Lab of Service Computing at Tongji University in 2010. Her research interests include Petri net and Web services.

Zhixiang Yin received her M.A. degree from Nanjing Normal University of SChina in 1996, and PhD. degree in the Huazhong University of Science and Technology in 2003. His research interests include Petri net and trusted software.

© 2012 NSP
Natural Sciences Publishing Cor.