Property Analysis of Refinement of Petri Net Based Representation for Embedded Systems

Chuanliang Xia*, Zhijun Zhang and Zhong Wang

School of Computer Science and Technology, Shandong Jianzhu University, Jinan, P.R. China

Abstract: Petri net refinement is a transformation by replacing a simple entity of a system with its functional and operational details. In general, the refined system may become incorrect even if the original system is correct because some of its original properties may have been lost or some undesired properties may have been created. For systems specified in an expended Petri net, this paper proposes conditions imposed on a kind of transition subnet refinement under which timing, functionality and reachability will be preserved. Such results can be applied nicely to solve design problems in intelligent building and also enhance the property-preserving approach and characterization-based approach for system verification.

Keyword: Transformation, Petri nets, Property analysis, Equivalence, Modelling.

1. INTRODUCTION

Transformations, such as refinements, synthesis, reductions, etc., are often applied on Petri nets in order to develop a correct design specification. One of the difficult tasks, in this process, is to verify that the transformed nets possess certain desirable properties.

There exists an approach for such purposes in the literature. In this approach, the original net is assumed to be satisfying some properties and the transformation is required to preserve these properties in the transformed net. This approach is called property preservation. The advantage of property preservation is that the transformed net is automatically correct without the need of further verification.

Refinement is an important transformation. This paper investigates a special type of refinement for PRES+[16] and its property preservation.

In the literature, As for refinement method, Lakos [1] defined three types of refinement for coloured Petri nets: node refinement, consisting in detailing the behaviour within a place or a transition; type refinement, which addresses the refinement of datatypes handled ; and subnet refinement which allows for the addition of a connected subnet. Völzer [2] studied the robustness of fairness notions under refinement of transitions and places in Petri nets. Choppy [3] used Coq theorem to prove the refinement relation between two Petri nets, both formally and automatically. For systems specified in pure ordinary Petri nets, Huang [4] proposed conditions imposed on several types of refinement under which 19 properties will be preserved. Xia [5, 6] proposed a kind of pp-type refinement and a kind of tt-type refinement for P/T Petri nets, and proved that these two kinds of refinements preserve some dynamic properties, such as boundednes, liveness and reversibility. Peuker [7] defined transition refinement for Algebraic Petri nets and studied how to prove that a replacement of a transition is a transition refinement. Cheung [8] proposed a refinement method for eliminating duplicate labels from a labeled net while preserving the original firing sequences (event sequences). Ahmad [9] introduced a refinement method for modeling parallel manufacturing system. Ding [10] proposed a stepwise refinement method which preserve language properties. Cabac [11] proposed a method to design a complex dynamic system at different levels of abstractions using refinements of net models. Chemaa [12] proposed an expressive object-oriented Petri net based algebra that succeeds in the complex composition of Web services, and gave a refinement operation which permits to replace certain operations of the service by more detailed ones. Köhler [13] described a special kind of Petri nets that can modify their structures via dynamic refinement of transitions. A reconfiguring process of a manufacturing system model is developed by using colored timed object-oriented Petri nets [14]. Petri nets, object-oriented methods and stepwise refinement ideas are integrated together in this model.

In order to improve verification efficiency, we proposed a set of reduction rules for PRES+ ([15]). In order to model large systems, a refinement method is needed. In this paper we will propose one kind of refinement for PRES+ , and prove that this refinement method will preserve some important properties, such as reachability, timing and functionality.

The rest of the paper is organized as follows. The basic definitions of PRES+ are presented in Section 2. Then, the refinement representation method of PRES+ and its property preservation are presented in Section 3. Afterwards, a modelling example is addressed in section 4, where we will use the refinement method to model a temperature controller of Fire Protection System (FPS) in intelligent building. Conclusions are finally discussed in Section 5.

*Address correspondence to this author at the School of Computer Science and Technology, Shandong Jianzhu University, Jinan, P.R. China; Tel: +86-531-86361302; Fax: +86-531-86366705; E-mail: chuanliang_xia@126.com

1874-4443/13 2013 Bentham Open

2. DEFINITIONS OF PRES+

In this section we will quickly review key definitions. A more general discussion on PRES+ can be found in [16].

Definition 2.1 [16] A PRES+ model is a five-tuple $N=(P,T,I,O,M_0)$ where,

$P=\{p_1,p_2,...,p_m\}$ is a finite non-empty set of places; $T=\{t_1,t_2,...,t_n\}$ is a finite non-empty set of transitions; $I\subseteq P\times T$ is a finite non-empty set of input arcs which define the flow relation between places and transitions; $O\subseteq T\times P$ is a finite non-empty set of output arcs which define the flow relation between transitions and places; M_0 is the initial marking of the net.

Definition 2.2 [16] The firing of an enabled transition $t\in T$, for a binding $b=(k_1,k_2,...,k_a)$ changes a marking M into a new marking M'. As a result of firing the transition t, the following occurs:

(i) Tokens from its pre-set ${}^\bullet t$ are removed, that is, $M'(p_i)=M(p_i)-\{k_i\}$ for all $p_i\in{}^\bullet t$;

(ii) One new token $k=<v,r>$ is added to each place of its post-set $t^\bullet$, that is, $M'(p)=M(p)+\{k\}$ for all $p\in t^\bullet$. The token value of k is calculated by evaluating the transition function f with token values of tokens in the binding b as arguments, that is, $v=f(v_1,v_2,...,v_a)$. The token time of k is the instant at which the transition t fires, that is, $r=tt^*$ where $tt^*\in[tt^-,tt^+]$.

(iii) The marking of places different from input and output places of t remain unchanged, that is, $M'(p)=M(p)$ for all $p\in P\setminus{}^\bullet t\setminus t^\bullet$.

3. DEFINITIONS OF EQUIVALENCE

The following definitions introduce basic concepts to be used when defining the notions of property preservation for systems modeled in PRES+.

Definition 3.1 [16] A marking M' is said to be reachable from M if there exists a transition $t\in T$ whose firing changes M into M'.

Definition 3.2 [16] The reachability set $R(N)$ of N is the set of all markings reachable from M_0; If $M\in R(N)$ and M' is immediately reachable from M, then $M'\in R(N)$.

Definition 3.3 [16] N_1 and N_2 are said to be equivalent if the following conditions are satisfied.

(i) There exist such bijections $f_{in}:inP_1\rightarrow inP_2$ and $f_{out}:outP_1\rightarrow outP_2$ that define one-to-one correspondences between in(out)-ports of N_1 and N_2;

(ii) The initial markings $M_{1,0}$ and $M_{2,0}$ satisfy

$M_{1,0}(p)=M_{2,0}(f_{in}(p))\neq\Phi\ \ \forall p\in inP_1$,

$M_{1,0}(q)=M_{2,0}(f_{out}(q))=\Phi\ \ \forall q\in outP_1$;

(iii) For every $M_1\in R(N_1)$ such that $m_1(p)=0$ $\forall p\in inP_1$, $m_1(s)=m_{1,0}(s)\ \ \forall s\in P_1\setminus inP_1\setminus outP_1$

there exists $M_2\in R(N_2)$ such that

$m_2(p)=0\ \ \forall p\in inP_2$,

$m_2(s)=m_{2,0}(s)\ \ \forall s\in P_2\setminus inP_2\setminus outP_2$,

$m_2(f_{out}(q))=m_1(q)\ \ \forall q\in outP_1$

and vice versa.

(iv) For every $<v_1,r_1>\in M_1(q)$, where $q\in outP_1$, there exists $<v_2,r_2>\in M_2(f_{out}(q))$ such that $v_1=v_2$, and $r_1=r_2$, and vice versa.

4. REFINEMENT OPERATION

In this section we propose one type subnet refinement operation for PRES+. This operation preserves equivalence. At the same time, desire properties, such as reachability, timing and functionality will be preserved.

Definition 4.1 [16] A transition $t\in T$ is an in-transition of $N=(P,T,I,O,M_0)$ iff $\bigcup_{p\in inP} p^\bullet=\{t\}$. A transition $t\in T$ is an out-transition of $N=(P,T,I,O,M_0)$ iff $\bigcup_{p\in outP} {}^\bullet p=\{t\}$.

Definition 4.2 Let $N=(P,T,I,O,M)$ and $N_0=(P_0,T_0,I_0,O_0,M_0)$ be two nets. If

(1) $P_0\subset P, T_0\subset T$ and $P_0\neq\Phi, T_0\neq\Phi$,

(2) $F_0=F\cap((P_0\times T_0)\cup(T_0\times P_0))$,

then N_0 is said to be a subnet of N.

Definition 4.3 A net $N_{tt}=(P_{tt},T_{tt},I_{tt},O_{tt},M_{tt,0})$ is said to be a transition subnet of N if and only if,

(i) N_{tt} is a subnet of N,

(ii) N_{tt} is connected, $\{t_{in},t_{out}\}\subseteq T_{tt}$ and t_{in} is the unique in-transition of N_{tt}, t_{out} is the unique out-transition of N_{tt},

(iii) inP_{tt} is the set of in-ports and $outP_{tt}$ is the set of out-ports of N_{tt},

(iv) $\forall t\in T_{tt}-\{t_{in}\}$, t is disabled in $M_{tt,0}$.

Supposition 4.1 In the net $N=(P,T,I,O,M_0)$, If $\tilde{t}\in T$ is replaced by a transition subnet $N_{tt}=(P_{tt},T_{tt},I_{tt},O_{tt},M_{tt,0})$, then

(i) There exists a bijection $f_{in}:{}^\bullet\tilde{t}\rightarrow inP_{tt}$;

(ii) There exists a bijection $f_{out}:\tilde{t}^\bullet\rightarrow outP_{tt}$;

(iii) $\forall p\in{}^\bullet\tilde{t}$, $M_0(p)=M_{tt,0}(f_{in}(p))$, $\tau(p)=\tau(f_{in}(p))$;

(iv) $\forall p \in \tilde{t}^{\bullet}$, $M_0(p) = M_{tt,0}(f_{out}(p))$ and $\tau(p) = \tau(f_{out}(p))$;

(v) $\forall < v_1, r_1 > \in M_1(q)$, $q \in \tilde{t}^{\bullet}$, $\exists < v_2, r_2 > \in M_{tt1}(f_{out}(q))$, $v_1 = v_2$, and vice versa;

(vi) $e_{\tilde{t}}^{-}$ of $\tilde{t}$ is equal to the lower bound of the execution time of N_{tt};

(vii) $e_{\tilde{t}}^{+}$ of $\tilde{t}$ is equal to the upper bound of the execution of N_{tt}.

We consider the net N. Transition subnet $N_{tt} = (P_{tt}, T_{tt}, I_{tt}, O_{tt}, M_{tt,0})$ is a refinement of $\tilde{t} \in T$.

Definition 4.4 Transition subnet refinement operation $\mathrm{Re}\, f_{tt}(\tilde{t}, N_{tt})$: the refinement net $N' = (P', T', I', O', M_0')$ is obtained from $N = (P, T, I, O, M_0)$ by using $N_{tt} = (P_{tt}, T_{tt}, I_{tt}, O_{tt}, M_{tt,0})$ to replace $\tilde{t}$ ($\tilde{t} \in T$), where

(i) $P' = P \cup (P_{tt} \setminus inP_{tt} \setminus outP_{tt})$;

(ii) $T' = T \cup T_{tt}$;

(iii) If $(p,t) \in I$ or $(p,t) \in I_{tt}$ and $p \notin inP_{tt}$,then $(p,t) \in I'$; If $(p,\tilde{t}) \in I$, then $(p, t_{in}) \in I'$;

(iv) If $(t,p) \in O$, or $(t,p) \in O_{tt}$ and $p \notin outP_{tt}$, then $(t,p) \in O'$; If $(\tilde{t},p) \in O$, then $(t_{out}, p) \in O'$;

(v) $\forall p \in P$, $M_0'(p) = M_0(p)$;

$\forall p \in P_{tt} - inP_{tt} - outP_{tt}$, $M_0'(p) = M_{1,0}(p)$.

Suppose that N' is obtained from N by transition subnet refinement $\mathrm{Re}\, f_{tt}(\tilde{t}, N_{tt})$.

Definition 4.5 Let $N_{\tilde{t}} = (P_{\tilde{t}}, T_{\tilde{t}}, I_{\tilde{t}}, O_{\tilde{t}}, M_{\tilde{t},0})$ be a subnet of N , where $P_{\tilde{t}} = {}^{\bullet}\tilde{t} \cup \tilde{t}^{\bullet}$, $T_{\tilde{t}} = \{\tilde{t}\}$, $I_{\tilde{t}} = \{(p,\tilde{t}) \mid p \in {}^{\bullet}\tilde{t}\}$, $O_{\tilde{t}} = \{(\tilde{t}, p) \mid p \in \tilde{t}^{\bullet}\}$, $M_{\tilde{t},0}(p) = M_0(p)$ $\forall p \in P_{\tilde{t}}$.

Definition 4.6 Let $N_{tt}' = (P_{tt}', T_{tt}', I_{tt}', O_{tt}', M_{tt,0}')$ be a subnet of N' . Where in N' , $P_{tt}' = {}^{\bullet}t_{in} \cup t_{out}^{\bullet} \cup \{P_{tt} \setminus inP_{tt} \setminus outP_{tt}\}$, $T_{tt}' = T_{tt}$, $I'_{tt} = \{(p, t_{in}) \mid p \in {}^{\bullet}t_{in}, t_{in} \in T_{tt}\} \cup I_{tt} \quad - \{(s, t_{in}) \mid s \in inP_{tt}\}$, $O_{tt}' = \{(t_{out}, p) \mid p \in t_{out}^{\bullet}, t_{out} \in T_{tt}\} \cup O_{tt}$

$-\{(t_{out}, s) \mid s \in outP_{tt}\}$

$$M_{tt,0}'(p) = \begin{cases} M_{tt,0}(p) & p \in P_{tt} \setminus inP_{tt} \setminus outP_{tt} \\ M_0(p) & p \in {}^{\bullet}t_{in} \cup t_{out}^{\bullet} \end{cases} .$$

Theorem 4.1 $N_{\tilde{t}}$ and N_{tt}' are equivalent.

Proof. By Supposition 4.1, there exists a bijection $g_{in} : {}^{\bullet}\tilde{t} \to {}^{\bullet}t_{in}$ (where ${}^{\bullet}t_{in}$ are the in-ports of N_{tt}'); There exists a bijection $g_{out} : \tilde{t}^{\bullet} \to t_{out}^{\bullet}$ ($t_{out}^{\bullet}$ are the out-ports of N_{tt}');

Suppose $M_{\tilde{t},0}(p) \neq \Phi$., $\forall p \in {}^{\bullet}\tilde{t}$ and $M_{\tilde{t},0}(q) = \Phi$ for all $q \in \tilde{t}^{\bullet}$. By Definition 4.3 and Supposition4.1, the initial markings $M_{\tilde{t},0}$ and $M_{tt,0}'$ such that $\forall p \in {}^{\bullet}\tilde{t}$, $M_{\tilde{t},0}(p) = M_{tt,0}'(g_{in}(p)) \neq \Phi$., $\forall q \in \tilde{t}^{\bullet}$, $M_{\tilde{t},0}(q) = M_{tt,0}'(g_{out}(q)) = \Phi$.

For every $M_{\tilde{t},1} \in R(N_{\tilde{t}})$ such that $m_{\tilde{t}}(p) = 0$ for all $p \in {}^{\bullet}\tilde{t}$, by Definition 4.4, there exists $M_{tt}' \in R(N_{tt}')$ such that $m_{tt}'(p) = 0$ for all $p \in {}^{\bullet}t_{in}$, $m_{tt}'(s) = m_{tt,0}'(s)$ for all $s \in P_{tt}' \setminus {}^{\bullet}t_{in} \setminus t_{out}^{\bullet}$, $\forall q \in \tilde{t}^{\bullet}$ $m_{tt}'(g_{out}(q)) = m_{\tilde{t}}(q)$.

By Definition 4.3 and Supposition4.1, for every $< v_1, r_1 > \in M_{\tilde{t}1}(q)$, $q \in \tilde{t}^{\bullet}$, $\exists < v_2, r_2 > \in M_{tt}'(g_{out}(q))$ such that $v_1 = v_2$.

Since $e_{\tilde{t}}^{-}$ of $\tilde{t}$ is equal to the lower bound of the execution time of N_{tt}, and $e_{\tilde{t}}^{+}$ of $\tilde{t}$ is equal to the upper bound of the execution of N_{tt}, then $r_1 = r_2$. By Definition 3.3, $N_{\tilde{t}}$ and N_{tt}' are equivalent.

Corollary 4.1 Suppose that N' is obtained from N by transition subnet refinement $\mathrm{Re}\, f_{tt}(\tilde{t}, N_{tt})$. Then N' and N are equivalent.

Proof. Since $N - N_{\tilde{t}}$ is equal to $N' - N_{tt}'$, by Theorem 4.1, N' and N are equivalent.

5. APPLICATIONS

In this section we apply the results of Section 4 to model an intelligent temperature control system of Fire Protection System (FPS) in intelligent building.

In Fig. (**1**), transition t_{Init}: system initial; t_{K1}: set the upper limit temperature T_{high}; t_{K2}: set the lower limit temperature T_{low}; t_{Modi1}: modify temperature T_{high}; t_{Modi2}: modify temperature T_{low}; T_{Exc} : exchange T_{high} for T_{low}; $t_{A/D}$: A/D transformation; t_{tran}: transform A/D value (0~1023) into temperature value (0~100); $t_{\Pr e-alarm}$: to deliver out pre-alarm signal; t_{Alarm} : to deliver alarm signal.

When temperature $30^o <= T < 60^o$, the pre-alarm signal is delivered to the alarm control panel. If temperature $T >= 60°$, the alarm signal is delivered to the alarm control panel.

In Fig. (**2**), t_{K3}, t_{K4}: press modify button; t_{add3}, t_{add4}: add number 1; If $a > 9$ ($b > 9$), then $a = 0$ ($b = 0$).

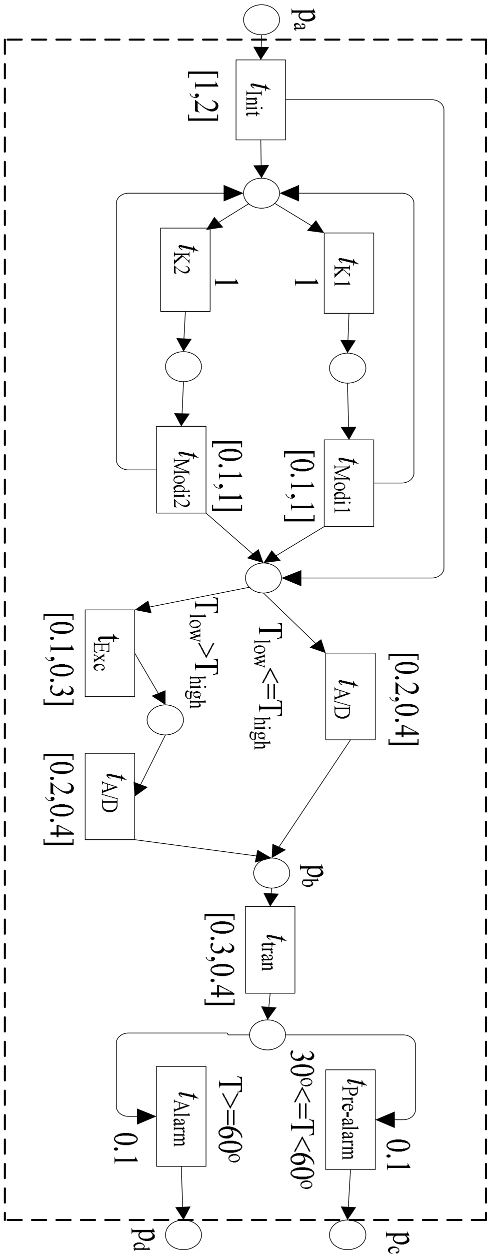

Fig. (1). The temperature control system.

The refined net N' is obtained by transition subnet refinement operation, i.e. using the transition subnet (Fig. **2**) to replace transition t_{Modi1} in N (Fig. **1**). By Corollary 4.1, N' and N are equivalent.

So, in N' the properties, such as reachability, timing and functionality of N are preserved. Note that other transitions, such as $t_{\text{mod}i2}$ and t_{tran}, can also be refined by transition subnet refinement operation.

6. CONCLUSIONS

In this paper we investigate property preservations of transition subnet refinement operation. Based on PRES+ model this paper proposes conditions under which reachability, timing and functionality will be preserved. These results can be applied nicely to solve design problems in intelligent building, manufacturing engineering and software engineering, and also release the designer's burden for having to provide different methods for individual properties. Further research is needed to investigate more general subnet refinement operations for PRES+ and their applications.

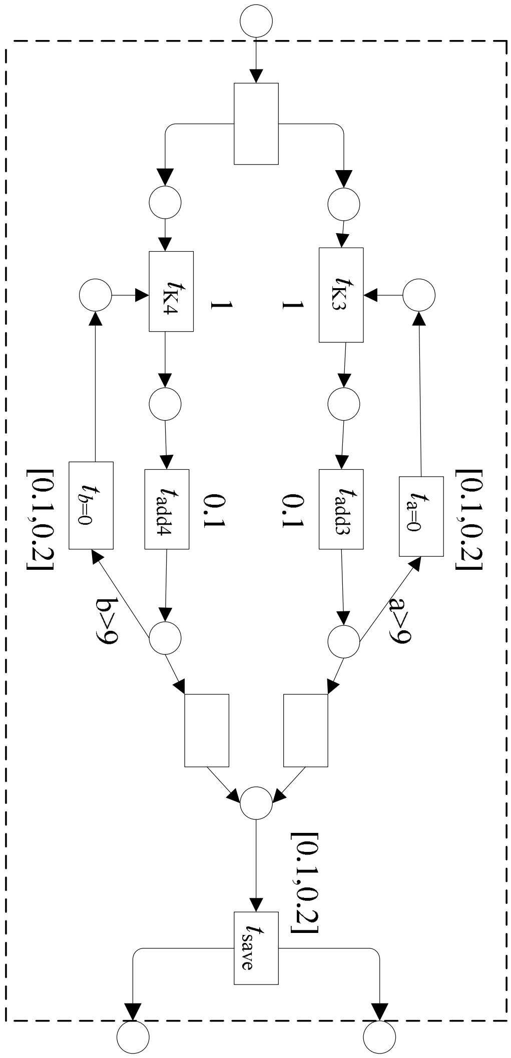

Fig. (2). The subnet.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflicts of interest.

ACKNOWLEDGEMENTS

This work was financially supported by the National Natural Science Foundation of China under Grant No. 60573012 and 60721061, the National Grand Fundamental Research 973 Program of China under Grant No. 2002cb312200, and Natural Science Foundation of Shandong Province under Grant No. ZR2012GQ010.

REFERENCES

[1] C. Lakos and G. Lewis, “Incremental state space construction of coloured Petri nets”, *Lect Notes Comput. Sci*., vol. 2075, pp. 263-282, 2001.

[2] H. Völzer, “Refinement-robust fairness”, *Lect. Notes Comput. Sci*., vol. 2421, pp. 547-562, 2002.

[3] C. Choppy, M. Mayero and L. Petrucci, “experimenting formal proofs of petri nets refinements”, *Electro Notes Theor. Comput. Sci.*, vol. 214, pp. 231-254, 2008.

[4] H. Huang, T.-y. Cheung and W. M. Mak, “Structure and behavior preservation by Petri-net-based refinements in system design”, *Theor Comput. Sci*, vol. 328 , pp. 245-269, 2004.

[5] C. Xia, “Analysis of properties of Petri synthesis net”, TAMC 2006, *Lect. Notes Comput. Sci.* 3959, pp. 576-587, 2006.

[6] C. Xia, “Property preservation by Petri-net-based refinements in system design”, In the 9th International Conferences for Young Computer Scientists (ICYCS 2008), pp. 240-246, 2008.

[7] S. Peuker, “Concurrency based transition refinement for the verification of distributed algorithms”, *Lect. Notes Comput. Sci.*, vol. 2472, pp. 430-454, 2003.

[8] K.S. Cheung, “Refinement of Petri-net-based system specification”, *Inform. Technol. Control*, vol. 35, no. 2, pp.137-143, 2006.

[9] F. Ahmad, H. Huang, X. Wang. “Analysis of Petri net model of parallel manufacturing processes”, *Inform. Sci.*, Vol. 181, pp. 5249-5266, 2011.

[10] Z. J. Ding, C. J. Jiang, M. C. Zhou and Y. Y. Zhang, “Preserving languages and properties in stepwise refinement-based synthesis of Petri nets”, *IEEE Trans Syst, Man, Cyberne Part A: Syst. Hum.*, vol. 38, no. 4, pp.791-801, 2008.

[11] L. Cabac, M. Duvigneau, D. Moldt, and H. Rölke, “Modeling dynamic architectures using nets-within-nets”, G. Giardo and P. Darondeau (Eds.): ICATPN 2005, LNCS 3536, pp.148-167, 2005.

[12] S. Chemaa, F. Bachtarzi and A. Chaoui, “A high-level Petri net based approach for modeling and composition of web services”, *Proc. Comput. Sci*., vol. 9, pp. 469-478, 2012.

[13] M. Köhler and H. Rölke, “Web service orchestration with super-dual object nets”, J. Kleijn and A. Yakovlev (Eds.): ICATPN 2007, LNCS 4546, pp. 363-280, 2007.

[14] X. Meng, “Modeling of reconfigurable manufacturing systems based on colored timed object-oriented Petri nets”, *J. Manuf. Syst.*, vol. 19, pp. 81-90, 2010.

[15] C. Xia, “Reduction rules for Petri net based representation for embedded systems”, *J. Front Comput. Sci. Technol.*, vol. 2, no. 6, pp. 614-626, 2008.

[16] L. A. Cortés, P. Eles and Z. Peng, “Modelling and formal verification of embedded systems based on a Petri net based representation”, *J. Syst. Arch.*, vol. 49, pp. 571-598, 2003.

Received: August 13, 2013 Revised: August 28, 2013 Accepted: August 28, 2013