Math. Struct. in Comp. Science (2014), vol. 24, iss. 5, e240514, 19 pages.
doi:10.1017/S0960129512000515

Liveness and boundedness analysis of Petri net synthesis†

CHUANLIANG XIA

School of Computer Science and Technology,
Shandong Jianzhu University, Jinan, China
and
State Key Laboratory of Computer Science, Institute of Software,
Academy of Sciences, Beijing, China
Email: chuanliang_xia@sdjzu.edu.cn

Received 20 July 2011; revised 18 August 2013

We provide motivation for and then study the synthesis of Petri nets. Synthesis can avoid the state exploration problem by guaranteeing correctness for the Petri net. We propose conditions to be imposed on a synthesis shared pb-type subnet for systems specified in Petri nets that ensure the preservation of the liveness and boundedness structural properties. Specifically, we propose a group of sufficient conditions, or both sufficient and necessary conditions, for liveness preservation and boundedness preservation. Possible applications of this synthesis method are illustrated through an example in the form of a flexible manufacturing system. These results are useful for studying the static and dynamic properties of Petri nets for analysing the properties of large complex systems.

1. Introduction

Subsystem sharing is a common and basic issue in system design. For example, in manufacturing engineering, plant and workstations are often shared by several processes as subsystems. For convenience, we can use the Petri net synthesis method to verify these subsystems.

Petri net based synthesis is a well-known approach to system design, which provides a conceptual foundation for synthesising a system from a set of component modules in such a way that the system can be effectively analysed for design correctness and consistency. We will begin by giving a brief review of some representative methods in the synthesis of Petri nets.

Agerwala and Choed-Amphai (1978) defined a kind of ST-net, which, under certain conditions, can be used to model some systems. Cheung (Krogh and Beck 1986) proposed a Petri-net-based synthesis methodology to resolve the use-case driven system design problem. The design objective for a number of concerns related to the sharing of resources, the finite capacity of system components and the need for system re-initialisation is to

† This work was financially supported by the National Natural Science Foundation of China under Grant Numbers 60573012, 60672180 and 60721061, the National Grand Fundamental Research 973 Program of China under Grant Number 2002cb312200, and CAS Key Laboratory of Computer Sciences (SYSKF0903).

obtain a robust system that is live, safe and reversible. Lorenz (Koh and DiCesare 1991) presented an algorithm for synthesising a finite place/transition Petri net (p/t-net) from a finite set of labelled partial orders (a finite partial language). The refinement and abstract representation method for Petri nets has been proposed (Badouel and Darondeau 2004), and is the key method for ensuring the synthesis net preserves well-behaved properties. Xia investigated the preservation of properties for a type of synthesised Petri net so that, subject to some constraints, liveness and boundedness are preserved after merging certain sets of subnets (Cheung *et al.* 2006).

Lorenz *et al.* (2008) presented a feedback control synthesis method to provide a systematic and easily implementable tool for specialists in the DES field. This method has the advantage of being applicable to both safe and non-safe PN models.

Xia (2006) proposed an approach for modelling Web service composition using Petri nets and based on OWL-S. In this approach, the boundedness and liveness properties of the Petri net models are analysed in order to guarantee the correctness of the composite Web service. An overall system Petri net model is obtained by synthesising individual modules satisfying system features such as production rates, buffer capacities and machine expected up, down or idle time (Xia 2008). Xia also investigated the transformation achieved by merging a set of Petri net subnets beyond asymmetric choice nets, and proposed conditions for it to preserve structural liveness (Vasiliu *et al.* 2009).

Kindler presented mining and synthesis algorithms that derive a Petri net model of a business process from a versioning log of a document management system (Ding *et al.* 2008). Carmona presented an algorithm for the synthesis of bounded Petri nets from transition systems (Ding *et al.* 2008). This algorithm has also been implemented in a tool. Bergenthum showed that VipTool can synthesise Petri nets from partially ordered runs and explained how, with the synthesis feature included, VipTool can be used for a stepwise and iterative formalisation and validation procedure for business process Petri net models (Tsinarakis *et al.* 2005).

Xia investigated one type of transformation and its property-preserving approach to verification (Xia 2005). He proposed a kind of sharing Single-Link subnet synthesis method, with conditions for preserving structural liveness of the synthesis net. In order to study the preservation of properties for the synthesis net, in the current paper we investigate another type of transformation and its property-preserving approach to verification. We propose a kind of sharing pb-type subnet synthesis method, which, subject to some conditions, ensures the preservation of the liveness and boundedness dynamic properties.

1.1. *Organisation of the paper*

We give basic definitions in Section 2, and then present the refinement and abstract representation method for Petri nets in Section 3. We obtain conditions that ensure the synthesis will preserve liveness and boundedness in Section 4, and present an application of the synthesis method for solving subsystem sharing problems in Section 5. Finally, we present our conclusions in Section 6.

2. Basic definitions

In this section, we will give a quick review of some key definitions – a more general discussion on Petri nets can be found in Murata (1989).

A weighted net is denoted by $N = (P, T, F, W)$ where:

— P is a non-empty finite set of places;
— T is a non-empty finite set of transitions with

$$P \bigcap T = \phi;$$

— F is a flow relation satisfying

$$F \subseteq (P \times T) \cup (T \times P);$$

— W is a weight function defined on the arcs, that is,

$$W : F \to \{1, 2, 3, \ldots\}.$$

We say

$$N = (P_1, T_1, F_1, W_1)$$

is a subnet of N if

$$\begin{aligned}
&P_1 \subset P\\
&T_1 = T\\
&P_1 \neq \varnothing\\
&T_1 \neq \varnothing\\
&F_1 = (P_1 \times T_1) \cup (T_1 \times P_1)\\
&W_1 = W \mid F_1 \qquad \text{(that is, } W_1 \text{ is the restriction of } W \text{ on } F_1\text{).}
\end{aligned}$$

A marking of a net $N = (P, T, F, W)$ is a mapping

$$M : P \to \{0, 1, 2, \cdots\}.$$

A Petri net is a pair (N, M_0), where N is a net and M_0 is the initial marking of N. A place p is said to be marked by M if $M(p) > 0$. A transition t is enabled or fireable at a marking M if for every $p \in t$, we have

$$M(p) \geqslant W(p, t).$$

A transition t may be fired if its enabled firing transition t changes the marking M to a new marking $M^{'}$, where $M^{'}$ is obtained by removing $W(p, t)$ tokens from each $p \in t$ and adding $W(t, p)$ tokens to every $p \in t^*$. This process is denoted by

$$M[t > M^{'}.$$

If

$$M[t_1 > M_1[t_2 > M_2 > \cdots M_{n-1}[t_n > M_n,$$

then $\sigma = t_1 \cdots t_n$ is called a firing sequence leading from M to M_n and is denoted by

$$M[\sigma > M_n.$$

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

We write $R(M_0)$ to denote the set of all markings reachable from the initial marking M_0.

A transition t is said to be live in (N, M_0) if for any $M \in R(M_0)$, there exists $M' \in R(M)$ such that t can be fired at M'. We say (N, M_0) is live if and only if every transition of N is live. A place p is said to be bounded in (N, M_0) if and only if there exists a constant k such that $M(p) \leqslant k$ for all $M \in R(M_0)$. We say (N, M_0) is bounded if every place of N is bounded.

Definition 2.1. A net

$$N_0 = (P_0, T_0, F_0, W_0)$$

is said to be a pb-type subnet of $N = (P, T, F, W)$ if and only if:

(1) N_0 is a subnet of$N$.
(2) ${}^{\bullet}T_0 \cup T_0^{\bullet} \cap (P - P_0) = \varnothing$.
(3) There exists a transition set

$$T_A \subseteq T - T_0$$

such that the subnet generated by P_0 and $T_0 \cup T_A$ forms a strongly connected state machine in N.
(4) There exists a directed path from every input place or initially maked place to every output place within N_0.

Definition 2.2. Suppose

$$N_1 = (P_1, T_1, F_1, W_1)$$
$$N_2 = (P_2, T_2, F_2, W_2)$$

are two Petri nets such that

$$P_1 \cap P_2 = P_0 \neq \varnothing$$
$$T_1 \cap T_2 = \varnothing.$$

Then $N = (P, T, F, W)$ is said to be a sharing synthesis net of N_1 and N_2 if

$$P = P_1 \cup P_2$$
$$T = T_1 \cup T_2$$
$$F = F_1 \cup F_2.$$

Definition 2.3. Suppose M_{10} is the initial marking of N_1 and M_{20} is the initial marking of N_2 such that for all $p \in P_0$ we have

$$M_{10}(p) = M_{20}(p).$$

Then the initial marking of the sharing synthesis net N is

$$M_0(p) = \begin{cases} M_{10}(p) & p \in P_1 \\ M_{20}(p) & p \in P_2. \end{cases}$$

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

The corresponding net system is

$$\Sigma_1 = (N_1, M_{10})$$
$$\Sigma_2 = (N_2, M_{20})$$
$$\Sigma = (N, M_0).$$

Σ is said to be the sharing synthesis net system of Σ_1 and Σ_2.

Definition 2.4. Suppose

$$N_1 = (P_1, T_1, F_1, W_1)$$
$$N_2 = (P_2, T_2, F_2, W_2)$$

are two Petri nets. Then $N = (P, T, F, W)$ is said to be a synthesis net of N_1 and N_2 with shared pb-type subnets if the following conditions are satisfied:

(1) We have

$$P_0 = P_1 \cap P_2 \neq \varnothing$$
$$T_0 = T_1 \cap T_2 \neq \varnothing.$$

(2) We have

$$P = P_1 \cup P_2$$
$$T = T_1 \cup T_2$$
$$F = F_1 \cup F_2.$$

(3) N_1 and N_2 share the pb-type subnet set N_0 defined by

$$N_0 = \{N_{pb1}, N_{pb2}, \cdots, N_{pbk}\},$$

where $N_{pbi}(i = 1, 2, \cdots, k)$ are pb-type subnets.

Definition 2.5. Suppose

$$\Sigma_1 = (N_1, M_{10})$$
$$\Sigma_2 = (N_2, M_{20})$$

are two Petri net systems, and suppose $\Sigma = (N, M_0)$ is such that

(1) N is a subnet of shared pb-type subnets of N_1 and N_2;
(2) for all $p \in P_0$ with $M_{10}(p) = M_{20}(p)$ we have M_0 is defined as

$$M_0(p) = \begin{cases} M_{10}(p) & p \in P_1 \\ M_{20}(p) & p \in P_2. \end{cases}$$

Then Σ is said to be the synthesis net system of the Σ_1 and Σ_2 shared pb-type subnets.

Definition 2.6. Suppose $\Sigma = (N, M_0)$ is a Petri net system and

$$P = P_1 \cup P_2$$
$$P_1 \cap P_2 = \varnothing.$$

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

Then (p_i, p_j) is said to be a place ordered pair of $\Sigma = (N, M_0)$ on P_1 if the following conditions are satisfied:

(1) $p_i, p_j \in P_1$ and $i \neq j$.

(2) If there exists $M_1 \in R(M_0)$ such that

$$M_1(p_i) = 0$$
$$M_1(p_j) > 0,$$

then for all $\sigma \in (T - p_j^\bullet)^*$ we have

$$M_1[\sigma > M_2$$
$$M_2(p_i) = 0.$$

Definition 2.7. Suppose (p_i, p_j) is a place ordered pair of Σ_1 on P_0 and (p_j, p_i) is a place ordered pair of Σ_2 on P_0. Then (p_i, p_j) is said to be an inter-reciprocal place ordered pair of Σ_1 and Σ_2 on P_0.

Definition 2.8. Suppose Σ is the sharing synthesis net system of Σ_1 and Σ_2, and $p_i, p_j \in P_0$. Then (p_i, p_j) is said to be a sharing place ordered pair if and only if:

(1) For all $M \in R(M_0)$ we have both

$$M(p_i) > 0$$
$$M(p_j) > 0.$$

(2) If $M[t' > M_1$ (where $t' \in T_1 \cap p_i^\bullet$), then for all

$$\sigma \in (T - T_1 \cap p_j^\bullet)^*$$

we have

$$M_1[\sigma > M_2$$

and for all

$$t \in T \cap p_j^\bullet$$

we have

$$\neg M_2[t > .$$

3. Refinement and abstract operations

In this section, we present a pb-type subnet refinement operation and a pb-type abstract operation. The two operations, which preserve boundedness and liveness, will be useful for proving some theorems in Section 4.

Definition 3.1 (pb-type subnet abstract operation). The Petri net

$$N' = \left(P', T', F', W'\right)$$

is obtained from the Petri net

$$N = (P, T, F, W)$$

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

by using a place P_0 to replace a pb-type subnet

$$N_0 = (P_0, T_0, F_0, W_0),$$

where:

(1) $P^{'} = (P - P_0) \cup \{p_0\}$;
(2) $T^{'} = T - T_0$;
(3) $F^{'} = (F - F_0 - (\{(t,p),(p,t) \mid t \in T - T_0, p \in P_0\} \cap F))$
$\cup\{(t,p_0) \mid t \in T - T_0, t^{\bullet} \cap P_0 \neq \varnothing\} \cup \{(p_0,t) \mid t \in T - T_0, t^{\bullet} \cap P_0 \neq \varnothing\}$;
(4) for all

$$t \in^{\bullet} P_0 \cup P_0^{\bullet} - T_0 - T_A,$$

we have

$$W^{'}(p_0,t) = |^{\bullet}\, t \cap P_0 \,|$$
$$W^{'}(t,p_0) = |^{\bullet}\, t \cap P_0 \,|;$$

(5) we have

$$M_0^{'}(p_0) = \sum_{p \in P_0} M_0(p)$$

and for $p \in P^{'} - \{p_0\}$

$$M_0^{'}(p) = M_0(p).$$

Definition 3.2 (pb-type subnet refinement operation). The Petri net

$$N = (P, T, F, W)$$

is obtained from the Petri net

$$N^{'} = \left(P^{'}, T^{'}, F^{'}, W^{'}\right)$$

by using a pb-type subnet

$$N_0 = (P_0, T_0, F_0, W_0)$$

to replace p_0. This operation is the inverse transformation of the pb-type subnet abstract operation, that is, $(N^{'}, M_0^{'})$ is transformed into (N, M_0).

Definition 3.3. The $(N^{'}, M_0^{'})$ obtained from (N, M_0) by a pb-type subnet abstract operation comprises the net $N^{'}$ and the marking $M_0^{'}$, where

$$M_0^{'}(p) = \begin{cases} \sum_{p \in P_0} M(p) & p = p_0 \\ M_0(p) & p \in P^{'} - \{p_0\}. \end{cases}$$

Lemma 3.1. Suppose $(N^{'}, M_0^{'})$ is obtained from (N, M_0) by a pb-type abstract operation. If there exist a fireable transition sequence $\sigma^{'}$ and marking $M^{'}$ such that

$$M_0^{'}[\sigma^{'} > M^{'},$$

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

then there exist a fireable transition sequence σ, corresponding to σ', and marking M such that

$$M_0[\sigma > M.$$

Proof. By the assumption, there exist a fireable transition sequence σ' and marking M' such that $M_0'[\sigma' > M'$. Suppose

$$\sigma' = \sigma_1' t_1 \sigma_2' t_2, \cdots, z_j \sigma_i' t_i, \cdots, z_j \cdots t_k \sigma_q',$$

where every

$$\sigma_i' \cap (p_0^\bullet \cup^\bullet p_0) = \varnothing,$$

every $z_i \in {}^\bullet p_0$ and every $z_i \in p_0^\bullet$. Then in (N, M_0), we have

$$t_i, z_i \in T_A \cup \{{}^\bullet P_0 \cup P_0^\bullet - T_0 - T_A\}.$$

By Definition 2.1, we have for all

$$x \in (P_0 \cap t_0^\bullet) \cup \{p \in P_0 \mid M_0(p) > 0\}$$

and for all

$$y \in {}^\bullet z_i \cap P_0, \quad i = 1, 2, \cdots n$$

there exists a path δ_i from x to y such that δ_i lies entirely within N_0. Since these paths lie within a connected state machine, they are all fireable sequences at M_1 if $M_1(p_\delta) > 0$, where $p_\delta \in \delta_i$ and $M_1 \in R(M_0)$, and every firing will preserve the number of tokens within P_0. In particular, some of them are fired so that every place in

$${}^\bullet z_i, \ i = 1, 2, \cdots n$$

gets a token eventually in (N, M_0). Let σ_i be such a firing sequence if a sequence in δ_i is fired. Then, the sequence

$$\sigma = \sigma_1' t_1 \sigma_2' t_2, \cdots, \sigma_i z_j \sigma_i' t_i, \cdots, \sigma_j z_j \cdots t_k \sigma_q',$$

is fireable. Since firing σ_i preserves the number of tokens, we have $M_0[\sigma > M$. □

Lemma 3.2. Suppose (N', M_0') is obtained from (N, M_0) by a pb-type abstract operation. If there exist a fireable transition sequence σ and marking M such that

$$M_0[\sigma > M,$$

then there exist a fireable transition sequence σ' (corresponding to σ) and marking M' such that

$$M_0'[\sigma' > M'.$$

Proof.

(1) If σ is the null sequence, it is obvious that $M = M_0$. Then, by Definitions 3.1 and 3.3, we have σ' is the null sequence and $M' = M_0'$.

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

(2) We now assume the proposition holds for every v, where $| v | \leqslant n$, that is, for such v and M_1, we have $M_0[v > M_1$ implies $M_0'[v' > M_1'$. Let $\sigma = vt$ and the marking M satisfy

$$M_0[v > M_1[t > M.$$

If $t \in T_0$, we have $\sigma' = v$.
By Definitions 3.1 and 3.3, we have

$$M' = M_1'$$
$$M_0'[\sigma' > M'.$$

If $t \in T - T_0$, we have $\sigma' = vt$.
By the above assumption, it is now sufficient to show that t is fireable at M'. By Defnition 3.3,

$$M_1'(p_0) = \sum_{p \in P_0} M_1(p)$$

and $M_1'(p) = M_1(p)$ for $p \in P - \{p_0\}$. Since t is fireable at M_1 in N, we have

$$M_1(p) \geqslant W(p,t)$$

for all $p \in^{\bullet} t$ in N. Hence

$$M'(p) \geqslant W'(p,t)$$

for all $p \in^{\bullet} t$ in N'. If

$$p \in {}^{\bullet}t \cup t^{\bullet} - \{p_0\},$$

we have

$$\begin{aligned} M'(p) &= M(p) \\ &= M_1(p) \pm 1 \\ &= M_1'(p) \pm 1 \\ M'(p_0) &= \sum_{p \in P_0} M(p) \\ &= M_1'(p_0) + W(t,p_0) - W(p_0,t). \end{aligned}$$

If

$$p \in P' - ({}^{\bullet}t \cup t^{\bullet}),$$

then

$$M'(p) = M(p) = M_1(p) = M_1'(p).$$

Hence,

$$M_0'[\sigma' > M_1'[t > M',$$

which completes the proof. □

Theorem 3.1. Suppose (N', M_0') is obtained from (N, M_0) by a pb-type abstract operation. Then (N', M_0') is bounded if and only if (N, M_0) is bounded.

Proof.

($\Rightarrow$) Suppose (N', M'_0) is bounded. For every reachable marking M of (N, M_0) we have, by Lemma 3.2, that M' is a reachable marking of (N', M'_0). Then, by Definition 3.1, for every place p in N, we have $M(p)$ is bounded by $M'(p)$.

($\Leftarrow$) Suppose (N, M_0) is bounded. For every $M' \in R(M'_0)$, since within N_0 there exists a directed path from every input place or initially marked place to every output place (Definition 2.1), we have, by Lemma 3.1, that $M'(p)$ is obviously bounded by $M(p)$ or $M(P_0)$. □

Theorem 3.2. Suppose (N', M'_0) is obtained from (N, M_0) by a pb-type abstract operation. Then (N', M'_0) is live if and only if (N, M_0) is live.

Proof.

($\Rightarrow$) Suppose (N', M'_0) is live. Then for every $\sigma \in L(N, M_0)$ and every $t \in T$, we have, by Lemma 3.2, that there exists $\sigma \in L(N', M'_0)$. Since (N', M'_0) is live, there exists $\sigma'_1 \in T'^*$ such that

$$\sigma' \sigma'_1 t \in L(N', M'_0).$$

Since there exists a directed path from every input place or initially marked place to every output place within N_0 (Definition 2.1), we have, by Lemma 3.1, that there exists $\sigma_1 \in T^*$ such that

$$\sigma \sigma_1 t \in L(N, M_0).$$

So, (N, M_0) is live.

($\Leftarrow$) Suppose (N, M_0) is live. Then for every

$$\sigma' \in L(N', M'_0)$$

and every $t \in T'$, since there exists a directed path from every input place or initially marked place to every output place within N_0 (Definition 2.1), by Lemma 3.1, there exists $\sigma \in L(N, M_0)$ corresponding to σ'. Since (N, M_0) is live, there exists $\sigma_1 \in T^*$ such that

$$\sigma \sigma_1 t \in L(N, M_0).$$

By Definition 3.1 and Lemma 3.2, we then have

$$\sigma' \sigma'_1 t \in L(N', M'_0).$$

So (N', M'_0) is live. □

4. Liveness and boundedness preservation of synthesis nets

In this section, we propose some conditions for the liveness preservation and boundedness preservation of synthesis nets. This is important for the property analysis of complex nets.

Theorem 4.1. Suppose Σ_1 and Σ_2 are two live and bounded Petri net systems, Σ is the synthesis net system of Σ_1 and Σ_2 with shared pb-type subnets. Let Σ'_1 and Σ'_2 be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations, and the corresponding place

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

set be P_0'. Let Σ' be the sharing synthesis net of Σ_1' and Σ_2'. If for all $p_1', p_j' \in P_0'$ we have (p_i', p_j') is not an inter-reciprocal place ordered pair of Σ_1' and Σ_2' on P_0', then Σ is live and bounded.

Proof. Suppose the pb-type subnet set is

$$N_0 = \{N_{p1}, N_{p2}, \cdots, N_{pk}\}.$$

Let Σ_1' and Σ_2' be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations. Let

$$P_0' = \left\{p_{01}', p_{02}', \cdots, p_{0k}'\right\}$$

be obtained from N_0 by a pb-type abstract operation. Since Σ_1 and Σ_2 are two live and bounded Petri net systems, by Theorems 3.1 and 3.2, Σ_1' and Σ_2' are two live and bounded Petri net systems. Since Σ_1' and Σ_2' are live and bounded, we have:

(1) In Σ_1', there exists an integer $k_1 > 0$ for all $p_1' \in P_1'$ and all $M_{11}' \in R(M_{10}')$ such that $M_{11}'(p_1') \leqslant k_1$.
(2) In Σ_2', there exists an integer $k_2 > 0$ for all $p_2' \in P_2'$ and all $M_{21}' \in R(M_{20}')$ such that $M_{11}'(p_1') \leqslant k_2$.

In Σ', we have for all $p' \in P'$ that

$$p' \in P_1' - P_0'$$

or

$$p' \in P_2' - P_0'$$

or

$$p' \in P_0'.$$

Considering these cases in turn:

— If $p' \in P_1' - P_0'$, then

$$\begin{aligned} M' \in R\left(M_0'\right), M'\left(p'\right) &\leqslant \max_{M_{11}' \in R\left(M_{10}'\right)} M_{11}'\left(p'\right) + \sum_{p'' \in P_0'} \max_{M_{21}' \in R\left(M_{20}'\right)} \left(M_{21}\left(p''\right)\right) \\ &\leqslant k_1 + \bar{k}k_2, \end{aligned}$$

(where $\bar{k}$ is the place number of P_0').

— If $p' \in P_2' - P_0'$, then for all $M' \in R\left(M_0'\right)$,

$$M'\left(p'\right) \leqslant k_2 + \bar{k}k_1.$$

— If $p' \in P_0'$, then for all $M' \in R\left(M_0'\right)$,

$$\begin{aligned} M'\left(p'\right) &\leqslant \max_{M_{11}' \in R\left(M_{10}'\right)} M_{11}'\left(p'\right) + \max_{M_{21}' \in R\left(M_{20}'\right)} \left(M_{21}\left(p''\right)\right) \\ &\leqslant k_1 + k_2. \end{aligned}$$

Let

$$k = (\bar{k} + 1)(k_1 + k_2).$$

Then for all $p^{'} \in P^{'}$ and all $M^{'} \in R(M_0^{'})$, we have $M^{'}(p^{'}) \leqslant k$, that is, $\Sigma^{'}$ is bounded.

We will now prove the liveness of $\Sigma^{'} = (N^{'}, M_0^{'})$. Suppose $\Sigma^{'} = (N^{'}, M_0^{'})$ is not live. Then there exist an $M_1^{'} \in R(M_0^{'})$ and a $t^{''} \in T^{'}$ for all $M^{'} \in R(M_1^{'})$ such that $\neg M^{'}[t^{''} >$. Now, it is obvious that

$$t^{''} \in \left\{ t^{'} \mid t^{'} \in p^{'\bullet}, \forall p^{'} \in P_0^{'} \right\}.$$

Without loss of generality, we can suppose $t^{''} \in T_1^{'}$. Since $\Sigma_1^{'}$ is bounded, we have that there exist $M_2^{'} \in R(M_1^{'})$, $p_j^{'} \in P_0^{'}$ and $t^{''} \in p_j^{'\bullet}$ such that

$$M_2^{'}\left(p_j^{'}\right) = 0,$$

and for all $p^{'} \in^{\bullet} t^{''} \cap P_1^{'}$,

$$M_2^{'}\left(p^{'}\right) \geqslant 1.$$

It is obvious that the resources of $p_j^{'}$ are used by $\Sigma_2^{'}$ and not given back to $p_j^{'}$. Since $\Sigma_2^{'}$ is live and bounded and $\Sigma_2^{'}$ has acquired the resources of $p_j^{'}$, if for all

$$t^{'} \in T_2^{'} \cap \left\{ t^{'} \mid t^{'} \in p^{'\bullet}, \forall p^{'} \in P_0^{'} : p^{'} \neq p_j^{'} \right\},$$

there exists a transition sequence $\sigma_1^{'} \in (T_2^{'})^*$ such that

$$M_2^{'}[\sigma_1^{'} > M^{''}$$
$$M^{''}[t^{'} >,$$

then there exist $t_j^{'''} \in^{\bullet} p_j^{'} \cap T_2^{'}$ and $\sigma_2^{'} \in (T_2^{'})^*$ such that

$$M^{''}[\sigma_2^{'} > M^{'''},$$

that is,

$$M^{'''}\left(p_j^{'}\right) \geqslant 1.$$

Hence,

$$M^{'''}[t^{''} > .$$

But this contradicts $\neg M[t^{''} >$ for all $M^{'} \in R\left(M_1^{'}\right)$.

So, there exists $p_i^{'} \in P_0^{'}$ with $p_i^{'} \neq p_j^{'}$ for all $\sigma^{'} \in (T_2^{'})^*$ and $M_2^{'}[\sigma^{'} > M_3^{'}$ such that

$$t^{'''} \in T_2^{'} \cap p_i^{'\bullet}$$

and

$$M_3^{'}[t^{'''} > .$$

Since for all $p^{'} \in^{\bullet} t^{'''} \cap P_2^{'}$ we have

$$M_3^{'}\left(p^{'}\right) \geqslant 1$$
$$M_3^{'}\left(p_j^{'}\right) = 0,$$

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

the resources of $p_i^{'}$ are used by $\Sigma_1^{'}$ and not given back to $p_i^{'}$, and the resources of $p_j^{'}$ are used by $\Sigma_2^{'}$ and not given back to $p_j^{'}$. This means that there exist $M_3^{'} \in R(M_0^{'})$ and $p_i^{'}, p_j^{'} \in P_0^{'}$ such that:

— $\Sigma_1^{'}$, which having used the resources of $p_i^{'}$, will give back the resources of $p_i^{'}$ after it has used the resources of $p_j^{'}$; and
— $\Sigma_2^{'}$, which having used the resources of $p_j^{'}$, will give back the resources of $p_j^{'}$ after it has used the resources of $p_i^{'}$.

By Definition 2.7, we have that $(p_i^{'}, p_j^{'})$ is an inter-reciprocal place ordered pair of $\Sigma_1^{'}$ and $\Sigma_2^{'}$ on $P_0^{'}$. However, this contradicts the fact that $(p_i^{'}, p_j^{'})$ is not an inter-reciprocal place ordered pair of $\Sigma_1^{'}$ and $\Sigma_2^{'}$ on $P_0^{'}$. Hence, $\Sigma^{'} = (N^{'}, M_0^{'})$ is live. Let Σ be obtained from $\Sigma^{'}$ by a pb-type refinement operation. In the process, $p_{01}^{'}, p_{02}^{'}, \cdots, p_{0k}^{'}$ should be replaced by $N_{p1}, N_{p2}, \cdots, N_{pk}$, respectively.

Finally, since $\Sigma^{'}$ is live and bounded, by Theorems 3.1 and 3.2, Σ is also live and bounded. □

Corollary 4.1. Suppose Σ_1 and Σ_2 are two live and bounded Petri net systems, and Σ is the synthesis net system of Σ_1 and Σ_2 with shared pb-type subnets. Let $\Sigma_1^{'}$ and $\Sigma_2^{'}$ be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations, and let the corresponding place set be $P_0^{'}$. Let $\Sigma^{'}$ be the sharing synthesis net of $\Sigma_1^{'}$ and $\Sigma_2^{'}$. If $\Sigma_1^{'}$ and $\Sigma_2^{'}$ have the same place ordered pair on $P_0^{'}$, then Σ is live and bounded.

Corollary 4.2. Suppose Σ_1 and Σ_2 are two live and bounded Petri net systems, and Σ is the synthesis net system of Σ_1 and Σ_2 with shared pb-type subnets. Let $\Sigma_1^{'}$ and $\Sigma_2^{'}$ be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations, and the corresponding place set be $P_0^{'}$. Let $\Sigma^{'}$ be the sharing synthesis net of $\Sigma_1^{'}$ and $\Sigma_2^{'}$. If $\Sigma_1^{'}$ and $\Sigma_2^{'}$ do not have a place ordered pair on $P_0^{'}$, then Σ is live and bounded.

Theorem 4.2. Suppose Σ_1 and Σ_2 are two live and bounded Petri net systems, and Σ is the synthesis net system of Σ_1 and Σ_2 with shared pb-type subnets. Let $\Sigma_1^{'}$ and $\Sigma_2^{'}$ be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations and the corresponding place set be $P_0^{'}$. Let $\Sigma^{'}$ be the sharing synthesis net of $\Sigma_1^{'}$ and $\Sigma_2^{'}$. If $M_0^{'}\left(p^{'}\right) = 0$ for all $p^{'} \in P_0^{'}$, then Σ is live and bounded.

Theorem 4.3. Suppose Σ_1 and Σ_2 are two live and bounded Petri net systems, and Σ is the synthesis net system of Σ_1 and Σ_2 with shared pb-type subnets. Let $\Sigma_1^{'}$ and $\Sigma_2^{'}$ be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations, and the corresponding place set be $P_0^{'}$. Let $\Sigma^{'}$ be the sharing synthesis net of $\Sigma_1^{'}$ and $\Sigma_2^{'}$. Then Σ is live and bounded if and only if $t^{'}$ is live for all

$$t^{'} \in \left\{t^{'} \mid t^{'} \in p^{'\bullet}, \forall p^{'} \in P_0^{'}\right\}.$$

Proof.

($\Leftarrow$) Suppose the pb-type subnet set is

$$N_0 = \{N_{p1}, N_{p2}, \cdots, N_{pk}\}.$$

Let Σ_1' and Σ_2' be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations. Now

$$P_0' = \left\{p_{01}', p_{02}', \cdots, p_{0k}'\right\}$$

is obtained from N_0 by pb-type abstract operations. Since Σ_1 and Σ_2 are both live and bounded Petri net systems, by Theorems 3.1 and 3.2, Σ_1' and Σ_2' are both live and bounded Petri net systems. Since t' is live for all

$$t' \in \left\{t' \mid t' \in p'^{\bullet}, \forall p' \in P_0'\right\},$$

we have for all $M' \in R(M_0')$ that there exists $M_1' \in R(M')$ such that $M'[t' >$. Without loss of generality, we can suppose

$$t' \in T_1' - \{t' \mid t' \in p'^{\bullet}, \forall p' \in P_0'\} \cap T_1'.$$

Since in Σ', we have

$$\{t' \mid t' \in p'^{\bullet}, \forall p' \in P_0'\}$$

is live, by Definitions 2.2 and 2.3, we have for all $M_1' \in R(M_0')$ there exists $\sigma_1' \in T'^*$ such that

$$M_1'[\sigma_1' > M_2'.$$

Suppose now that M_{12}' is the projection of M_2' on Σ_1'. Then there exists $\sigma_2' \in T_1'^*$ such that

$$M_{12}'[\sigma_2' > M_3'$$
$$M_3'[t' > .$$

Hence, Σ' is live and bounded.

Now let Σ be obtained from Σ' by pb-type refinement operation. In the process,

$$p_{01}', p_{02}', \cdots, p_{0k}'$$

should be replaced by

$$N_{p1}, N_{p2}, \cdots, N_{pk},$$

respectively. Since Σ' is live and bounded, Σ is also live and bounded by Theorems 3.1 and 3.2.

($\Rightarrow$) Since Σ is live, it is obvious that t' is also live for all

$$t' \in \left\{t' \mid t' \in p'^{\bullet}, \forall p' \in P_0'\right\},$$

which completes the proof. □

Theorem 4.4. Suppose Σ_1 and Σ_2 are two live and bounded Petri net systems, and Σ is the synthesis net system of Σ_1 and Σ_2 with shared pb-type subnets. Let Σ_1' and Σ_2' be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations, and the corresponding place set be P_0'. Let Σ' be the sharing synthesis net of Σ_1' and Σ_2'. Let (p_i', p_j') be an inter-reciprocal place ordered pair of Σ_1' and Σ_2' on P_0'. Then, if (p_i', p_j') and (p_j', p_i') are sharing place ordered pairs of Σ_1' and Σ_2', respectively, then Σ is live and bounded.

Proof. Suppose the pb-type subnet set is

$$N_0 = \{N_{p1}, N_{p2}, \cdots, N_{pk}\}.$$

Let $\Sigma_1^{'}$ and $\Sigma_2^{'}$ be obtained from Σ_1 and Σ_2, respectively, by pb-type abstract operations. Then

$$P_0^{'} = \left\{p_{01}^{'}, p_{02}^{'}, \cdots, p_{0k}^{'}\right\}$$

is obtained from N_0 by pb-type abstract operations. Since Σ_1 and Σ_2 are both live and bounded Petri net systems, $\Sigma_1^{'}$ and $\Sigma_2^{'}$ are also live and bounded Petri net systems by Theorems 3.1 and 3.2.

Now, for every inter-reciprocal place ordered pair $(p_i^{'}, p_j^{'})$ of $\Sigma_1^{'}$ and $\Sigma_2^{'}$ on $P_0^{'}$, if

$$t^{''} \in p_i^{'\bullet} \cap T_1^{'}$$
$$t^{'''} \in p_j^{'\bullet} \cap T_2^{'},$$

then $t^{''}$ and $t^{'''}$ do not fire for all $M^{'} \in R(M_0^{'})$. By the liveness of $\Sigma_1^{'}$ and $\Sigma_1^{'}$, if there exists $t^{''} \in p_i^{'\bullet} \cap T_1^{'}$ for all $M^{'} \in R(M_0^{'})$ such that $M_1^{'}[t^{''} > M_2^{'}$, then for all $t^{'} \in p_j^{'\bullet} \cap T_1^{'}$, there exists $M_3^{'} \in R(M_0^{'})$ such that $M_3^{'}[t^{'} >$, that is, $t^{'}$ is live.

For the same reason, if $t^{'''} \in p_j^{'\bullet} \cap T_2^{'}$ and $M_1^{'}[t^{'''} > M_2^{'}$ for all $M^{'} \in R(M_0^{'})$, then $t^{'}$ is live for all $t^{'} \in p_i^{'\bullet} \cap T_2^{'}$.

Now, for all $p_i^{'}, p_j^{'} \in P_0^{'}$, if $(p_i^{'}, p_j^{'})$ is not an inter-reciprocal place ordered pair of Σ_1 and Σ_2 on $P_0^{'}$, by Corollary 4.1, we have $t^{'}$ is live for all $t^{'} \in p_k^{'\bullet}(k = 1, 2)$. Hence, $t^{'}$ is live for all

$$t^{'} \in \left\{t^{'} \mid t^{'} \in p^{'\bullet}, \forall p^{'} \in P_0^{'}\right\}.$$

Hence, by the proofs of Theorems 4.1 and 4.3, $\Sigma^{'}$ is live and bounded. So, let Σ be obtained from $\Sigma^{'}$ by a pb-type refinement operation. In the process,

$$p_{01}^{'}, p_{02}^{'}, \cdots, p_{0k}^{'}$$

should be replaced by

$$N_{p1}, N_{p2}, \cdots, N_{pk},$$

respectively. Since $\Sigma^{'}$ is live and bounded, Σ is also live and bounded by Theorem 3.1 and Theorem 3.2, and the proof is complete. □

5. Applications

In this section, we apply the results of Section 4 to design a flexible manufacturing system using two other flexible manufacturing systems that share a subsystem.

We consider two manufacturing systems, each of which consists of one machining centre. The two systems run as follows:

System A: In this case, the intermediate parts are machined by machine M_1 in the machining centre. Each part is fixed to a pallet and loaded into the machine M_1 by robot R. After processing, robot R unloads the final product, releases it from the pallet and then returns the pallet.

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

System B: In this case, the raw parts are machined first by machine M_2 and then by machine M_3 . Each part is fixed to a pallet and loaded into the machine by robot R. After processing, robot R unloads the intermediate part from M_2 into buffer B. At machine M_3, the intermediate parts are automatically loaded into M_3 and processed. When M_3 has finished processing a part, robot R unloads the final product, releases it from the pallet and then returns the pallet.

In order to share resources and enhance efficiency, a manufacturing system can be obtained from system A and system B by a synthesis operation. In order to specify the manufacturing system using Petri nets, each operation process is abstracted to a single place and each transition represents the start of and/or completion of a process. We will first give the Petri-net based models of Systems A and B, respectively, and then a synthesis net system obtained from Systems A and B by a synthesis operation with a shared pb-type subnet. Finally, we will analyse the preservation properties of the synthesis operation.

The Petri-net based model $\Sigma_1 = (N_1, M_{10})$ of System A is shown in Figure 1, where the meanings of the places are:

— p_{11}: pallet and intermediate parts are available;
— p_{12}: request robot R;
— p_{13}: acquire R, and R loads the pallet into machine M_1;
— p_{14}: intermediate parts are processed in M_1;
— p_{15}: machine M_1 is available;
— p_{16}: request robot R;
— p_{17}: robot R unloads a final product and returns the pallet;
— $p_{0i}(i = 1, 2, 3)$: robot R is available;

and the meanings of the tranistions are

— t_{11}: start activity p_{13};
— t_{12}: complete p_{13} and start activity p_{14};
— t_{13}: complete p_{14} and start activity p_{17};
— t_{14}: complete p_{17};
— $t_{0i}(i = 1, 2, 3)$: intermediate processing on a robot before passing it from one process to another.

The Petri-net based model $\Sigma_2 = (N_2, M_{20})$ of System B is shown in Figure 2, where the meanings of the places are:

— p_{21}: pallet and raw parts are available;
— p_{22}: machine M_2 loads and processes a part;
— p_{23}: machine M_2 is available;
— p_{24}: intermediate parts are available;
— p_{25}: robot R loads an intermediate part into buffer B;
— p_{26}: buffer B is available;
— p_{27}: intermediate parts are available;
— p_{28}: machine M_3 processes an intermediate part, R unloads a final product from M_3, releases and returns the pallet;
— p_{29}: macnine M_3 is available;

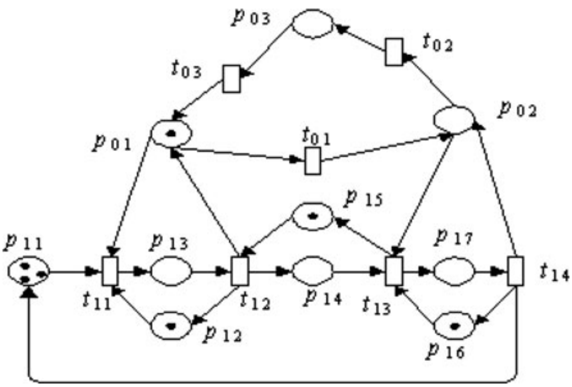

Fig. 1. Petri net system $\Sigma_1 = (N_1, M_{10})$

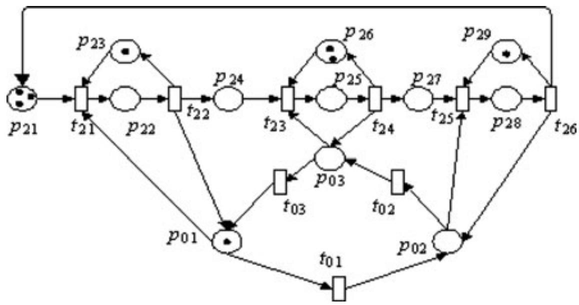

Fig. 2. Petri net system $\Sigma_2 = (N_2, M_{20})$

and the meanings of the tranistions are

— t_{21}: starts activity p_{22};
— t_{22}: completes p_{22} and start activity p_{24};
— t_{23}: completes p_{24} and start activity p_{25};
— t_{24}: completes p_{25} and start activity p_{27};
— t_{25}: completes p_{27} and start activity p_{28};
— t_{26}: completes p_{28}.

Systems A and B both contain a pb-type subnet N_0, which is generated by

$$\{p_{01}, p_{02}, p_{03}, t_{01}, t_{02}, t_{06}\}.$$

In order to save resources (such as the robot) and enhance efficiency, a synthesis net system $\Sigma = (N, M_0)$ (see Figure 3) can be obtained from System A

$$\Sigma_1 = (N_1, M_{10})$$

and System B

$$\Sigma_2 = (N_2, M_{20})$$

by a synthesis operation with the shared pb-type subnet. Since Σ_1 and Σ_2 are live and bounded net systems, the synthesis net system $\Sigma = (N, M_0)$ is also live and bounded by Theorem 4.3.

https://doi.org/10.1017/S0960129512000515 Published online by Cambridge University Press

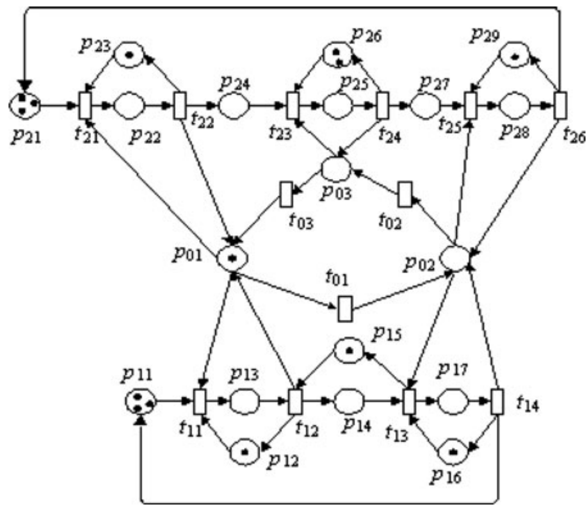

Fig. 3. The synthesis net system $\Sigma = (N, M_0)$

6. Conclusions

In this paper, we have investigated the preservation of properties for synthesis Petri nets. We have proposed a refinement and abstract representation method for Petri nets. Given some additional constraints, liveness and boundedness are preserved after merging some sets of pb-type subnets of Petri nets. As a consequence, this result can be usefully applied to solve some subsystem sharing problems in software engineering and manufacturing engineering. In comparison to most existing methods, which are only applied to state machines, marked graphs or AC nets to solve resource-sharing problem, our method is applicable to a wider class of Petri nets than AC nets to solve subsystem sharing problems. Further research is needed to determine how to extend the results obtained in this paper to more general types of nets.

References

Agerwala, T. and Choed-Amphai, Y. (1978) A synthesis rule for concurrent systems. *Proceedings of the 15th Design Automation Conference* 305–311.

Badouel, E. and Darondeau, Ph. (2004) Representing control: a study of the CPS transformation. *Information and Computation* **193** (2) 117–135.

Cheung, K. S., Cheung, T. Y. and Chow, K. O. (2006) A Petri-net-based synthesis methodology for use-case-driven system design. *The Journal of Systems and Software* **79** (7) 772–790.

Ding, Z. J., Wang, J. L. and Jiang, C. J. (2008) An approach for synthesis Petri nets for modeling and verifying composite web service. *Journal of Information Science and Engineering* **24** (10) 1309–1328.

Koh, K. and DiCesare, F. (1991) Modular transformation methods for generalised petri nets and their application to automated manufacturing systems. *IEEE Transactions on Systems, Man and Cybernetics* **21** (6) 1512–1522.

Krogh, B. H. and Beck, C. (1986) Synthesis of place transition nets for simulation and control of manufacturing systems. In: Geering, H. and Mansour, M. (eds.) *Large-Scale Systems: Theory and Applications*, Pergamon Press 583–588.

Lorenz, R., Bergenthum, R., Desel, J. and Mauser, S. (2008) Synthesis of Petri nets from finite partial languages. *8th International Conference of Concurrency to System Design* 170–179.

Murata, T. (1989) Petri nets: properties, analysis and applications. *Proceedings of IEEE* **77** (4) 541–580.

Tsinarakis, G. J., Tsourveloudis, N. C. and Valavanis, K. P. (2005) An approach for synthesis Petri nets for modeling and verifying composite web service. *Journal of Intelligent manufacturing* **16** (1) 67–92.

Vasiliu, A. I., Dideban, A. and Alla, H. (2009) Control synthesis for manufacturing systems using non-safe Petri nets. *CEAI* **11** (2) 43–50.

Xia, C. (2005) Structural liveness preservation of Petri synthesis net. *Journal of Computational Information Systems* **13** (12) 4485–4492.

Xia, C. (2006) Analysis of properties of Petri synthesis net. *Proceedings of the 3rd International Conference on Theory and Applications of Models of Computation* 576–587.

Xia, C. (2008) Property preservation by Petri net synthesis in system design. *Proceedings of International Technology and Environmental System Sciences* 6–11.