Analysis and Application of Petri Subnet Reduction

Chuanliang Xia 1,2
1. School of Computer Science and Technology, Shandong Jianzhu University, Jinan, China
2. State Key Laboratory of Computer Science, Institute of Software, Academy of Sciences, Beijing, China
Email: chuanliang_xia@126.com

Abstract—We motivate and study the subnet reduction of Petri nets. Subnet reduction can avoid the state exploration problem by guaranteeing the correctness in the Petri net. For systems specified in Petri nets, this paper proposes two subnet reduction methods. One major advantage of these reduction methods is that the resultant ordinary Petri net is guaranteed to be live, bounded and reversible. A group of sufficient conditions or sufficient and necessary conditions of liveness preservation, boundedness preservation and reversibility preservation are proposed. A flexible manufacturing system has been verified. These results are useful for studying the static and dynamic properties of Petri nets, analyzing properties for large complex system.

Index Terms—Petri nets; reduction; property analysis; liveness; system verification

I. INTRODUCTION

Petri nets are well known for their graphical and analytical capabilities in specification and verification, especially for concurrent systems. Many properties can be analytically defined and many techniques are available for development and verification. In particular, the approach based on property-preserving transformations will be described in more detail in this paper.

Usually, a design may be subject to many transformations, such as reduction, synthesis, refinement, etc. Research in reduction methods began with simple pattern modifications on Petri nets [1, 2]. Desel [2] showed that a live and safe FC (free choice) net without frozen tokens can be reduced either to a live and safe marked graph or to a live and safe state machine. A well-known result is the preservation of well-formedness and Commoner's property under the merge of places within a free choice net or an asymmetric-choice net [3].

In order to improve the PRES+ net verification efficiency, Xia [4] proposed a set of reduction rules, these reduction rules preserve total-equivalence. Murata [5] presented six reduction rules to reduced ordinary Petri nets, these rules preserved liveness, safeness and boundedness. Sloan [6] introduced a notion of equivalence among time Petri nets, and proved that their reduction rules yield equivalent net. This notion of equivalence guarantees that crucial timing and concurrency properties were preserved. Most reductions are quite specific, such as merging a few places or transitions [7, 8], reducing individual places or transitions [9, 10] or very specific subnets. Esparza [11] provided reduction rules for LTL model-checking of

l-safe Petri nets. In order to improve the analysis efficiency, Shen [12] reduced a large digital system to a simpler one by using three kinds of reduction rules. Based on Delay Time Petri Net (DTPN), Jiang [13] transformed a Time Petri Net (TPN) component to DTPN model in order to preserve such properties as synchronization, conflict and concurrency during the reduction. Huang [14] proposed some new rules to detect the existence of structural conflicts.

As for synthesis method, Franceschinis [15] presented the application of a compositional modeling methodology to the re-engineering of stochastic Well Formed Net (SWN) models of a contact center, the advantages are that this approach, based on the definition of classes and instances of submodels, can provide to the application of SWN to complex case studies. An effective solution to the net synthesis problem for path-automatic specifications is presented in [16]. A module synthesis method of EN-system was presented in [17]. The refinement and abstract representation method of Petri net is proposed [18], which is the key method to ensure the synthesis net preserving the well-behaved properties. Conditions of structural liveness preservation of a kind of Petri synthesis net is proposed [19].


This paper investigates one type of transformations and its property-preserving approach for verification. Two kinds of subnet reduction methods are proposed. Conditions of liveness, boundedness and reversibility preservation of ordinary Petri reduction net are proposed.

This paper is organized as follows. Section 2 presents preliminaries. Section 3 investigates p-subnet reduction method. Section 4 studies t-subnet reduction method. Section 5 gives an example of manufacturing system verification. Section 6 concludes this paper.

II. PRELIMINARIES

In this section we will quickly review key definitions. A more general discussion on Petri nets can be found in [5, 21].

A weighted net is denoted by \( N = (P, T; F, W) \) where \( P \) is a non-empty finite set of places, \( T \) is a

© 2011 ACADEMY PUBLISHER
doi:10.4304/jcp.6.8.1662-1669
non-empty finite set of transitions with $P \cap T = \emptyset$, $F \subseteq (P \times T) \cup (T \times P)$ is a flow relation and $W$ is a weight function defined on the arcs, i.e., $W : F \to \{1,2,3,...\}$. $N_i = (P_i, T_i; F_i, W_i)$ is called a subnet of $N$ if $P_i \subset P$, $T_i = T$, $P_i \neq \phi$, $T_i \neq \phi$, $F_i = (P_i \times T_i) \cup (T_i \times P_i)$ and $W_i = W | F_i$, i.e., the restriction of $W$ on $F_i$.

A marking of a net $N = (P, T; F, W)$ is a mapping $M : P \to \{0,1,2,\ldots\}$. A Petri net is a couple $(N, M_0)$, 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 $M(p) \geq 0$. A transition $t$ is enabled or firable at a marking $M$ if for every $p \in t$, $M(p) \geq W(p, t)$. A transition $t$ may be fired if it is enabled. Firing transition $t$ results in changing the marking $M$ to a new marking $M'$, where $M'$ is obtained by removing $W(p, t)$ tokens from each $p \in t$. The process is denoted by $M[t > M']$. If $M[t_1 > M][t_2 > \ldots M[n-1][t_n > M_n]$ then $\sigma = t_1\ldots t_n$ is called a firing sequence leading from $M$ to $M_n$ and is denoted as $M[\sigma > M_n]$. $R(M_0)$ denotes the set of all markings reachable from the initial marking $M_0$.

A transition $t$ is said to be live in $(N, M_0)$ iff for any $M \in R(M_0)$, there exists $M' \in R(M)$ such that $t$ can be firing at $M'$. $(N, M_0)$ is said to be live iff every transition of $N$ is live. A place $p$ is said to be bounded in $(N, M_0)$ iff there exists a constant $k$ such that $M(p) \leq k$ for all $M \in R(M_0)$. $(N, M_0)$ is bounded iff every place of $N$ is bounded.

III. p-SUBNET REDUCTION METHOD

In this section we present p-subnet reduction operation. This operation preserves boundedness, liveness and reversibility.

**Definition 3.1** A net $N_0 = (P_0, T_0; F_0, W_0)$ is said to be a p-subnet of $N = (P, T; F, W)$ iff,

1. $N_0$ is a subnet of $N$,
2. $T_0 \cup T^* \subseteq P_0$,
3. $N_0$ is connected, $\{p_1, p_o\} \subseteq P_0$ and $p_i$ is the only input place of $N_0$, $p_o$ is the only output place of $N_0$.

Supposition 3.1 A p-subnet satisfies:

1. $p_i$ is the only place which can contain the initial marking (token(s)).
2. In a process (tokens from outside flow into $p_i$, pass $N_0$ and then flow out from $p_o$), the number of tokens flowing into $p_i$ is equal to the number of tokens flowing out from $p_o$.

An example of p-subnet is illustrated in Fig.3.1.

**Definition 3.2** p-subnet reduction operation: a reduced net $N' = (P', T', F', W')$ is obtained from original Petri net $N = (P, T; F, W)$ by using $\bar{p}$ to replace a p-subnet $N_p = (P_p, T_p; F_p, W_p)$, where

1. $P' = P \cup \{\bar{p}\} - P_p$,
2. $T' = T - T_p$,
3. $F' = F - (t, p) | t \in P_p$ - $(p_o, t) | t \in P_o^*$

**Definition 3.3** $(N', M')$ obtained from $(N, M_0)$ by p-subnet reduction operation comprises net $N'$ and marking $M_0'$ where

$$M_0' = \begin{bmatrix} M_{(N_p, M_p^{(0)}, 0)} & M_{(N_p, M_p^{(0)}, n)} \end{bmatrix}$$

$(N_p, M_p^{(0)})$ is obtained from $M$ by deleted the vector corresponding to $P_p$.

**Definition 3.4** A net $(\bar{N}_p, \bar{M}_p)$ is said to be a closed p-subnet if adding a transition set $T_p = \{t_p \mid t_p$ corresponding to $t \in \bar{p}^*\}$ and arc set $\{(p_o, t_p), (t_p, p_i) \mid t \in T_p^*\}$ to $(N_p, M_p^{(0)})$, and preserving the marking of $(N_p, M_p^{(0)})$.

Note that in this section, let $(\bar{N}_p, \bar{M}_p)$: the original net;

$N_p = (P_p, T_p; F_p, W_p)$: the p-subnet;

$(N_p, M_p^{(0)})$: the p-subnet system;

$(\bar{N}_p, \bar{M}_p)$: the closed p-subnet system.
\((N, M_0)\): the reduced net.

**Theorem 3.1** Suppose that \((N, M_0)\) is obtained from \((N', M_0')\) by \(p\)-subnet reduction operation. Then \((N', M_0')\) is bounded iff \((N, M_0)\) and \((\overline{N}_p, \overline{M}_{p_0})\) are bounded.

**Proof.** (1) Since \((N, M_0)\) is bound, then \(\forall p \in P\), \(\exists k_i > 0\), such that \(M(p) \leq k_i\), \(\forall M \in R(M_0)\). Obviously, \(\forall p \in P - \{\overline{p}\}, M_{(p,p)}(p) \leq k_i\). Since \((\overline{N}_p, \overline{M}_{p_0})\) is bound, then \(\forall p \in P_p\), \(\exists k_2 > 0\), such that \(M_p(p) \leq k_2\), \(\forall M_p \in R(M_{p_0})\). Let \(k = k_1 + k_2\), by Supposition 3.1, \(\forall p \in P'\), \(M'(p) \in [M_{(p,p)}(p) \leq k\), \(\forall M' \in R(M_0')\), so \((N', M_0')\) is bound.

(2) Suppose that \((N, M_0)\) is unbound, then \(\exists p \in P\), \(\exists k > 0\), \(\exists M \in R(M_0)\) and \(M(p) > k\). By Supposition 3.1 and Definition 3.1-3.4, \(\forall k > 0\), \(\exists M' \in R(M_0)\) and \(M'(p) > k\). This contradicts with the fact that \((N', M_0')\) is bounded.

**Theorem 3.2** Suppose that \((N, M_0)\) is obtained from \((N', M_0')\) by \(p\)-subnet reduction operation. If \((N', M_0')\) is live and \(p \in p(p \in P) \land (M'_0(p) > 0)\), then \((N, M_0)\) and \((\overline{N}_p, \overline{M}_{p_0})\) are live.

**Proof.** Suppose that \(\forall M'_0'' \in R(M_0')\), \((N, M_0)\) (obtained form \((N', M_0'')\)) by \(p\)-subnet reduction operation) is not live, i.e. \(\exists M \in R(M_0)\), \(\exists t \in T\), \(\forall M' \in R(M)\), \(-\overline{M}(t)\). Since \((N', M_0')\) is live, \(M_{(p,p)}(0)\) is the projection of \(M''_0\) on \(P - \{\overline{p}\}\), then 
\[M_{(p,p)}(\sigma \sigma > M_{(p,p)}(\overline{M}_{(p,p)}))\sigma, \sigma, T\] (where \(\sigma\) and \(\overline{\sigma}\) are firing transition sequences of \((N, M_0)\)). If transition (or transition step) \(\sigma\) is added to the firing transitions, we get \(\sigma', \sigma'' \in T''\) such that \(M''_0\sigma > M''_0\sigma > \overline{M}''\) and \(M_{(p,p)}\) is the projection of \(M''_0\) on \(P - \{\overline{p}\}\). Then according to \(M''_0 \in R(M_0'')\), \(\exists t' \in T''\) (where \(t' = t\)), such that \(\forall \overline{M}'' \in R(M'')\) according to \(\forall \overline{M}_{(p,p)}\). Since \(-\overline{M_{(p,p)}}(t')\), then \(-\overline{M''}(t')\). This contradicts with the fact that \((N', M_0')\) is live. Then \(\exists M''_0, M'''_0 \in R(M_0')\), such that \((N, M_0)\) and \((\overline{N}_p, \overline{M}_{p_0})\) are live. Since \(p \in p(p \in P) \land (M'_0(p) > 0)\), then \((N, M_0)\) and \((\overline{N}_p, \overline{M}_{p_0})\) are live.

**Theorem 3.3** Suppose that \((N, M_0)\) is obtained from \((N', M_0')\) by \(p\)-subnet reduction operation. If \((N, M_0)\) and \((\overline{N}_p, \overline{M}_{p_0})\) are live, then \((N', M_0')\) is live.

**Proof.** In \((N', M_0')\), \(\forall t' \in T'\), then \(t' \in T\) or \(t' \in T_p\). \(\forall M' \in R(M_0')\), let \(M' = [M_{(p,p)}, M_p]\), by Supposition 3.1, we get \(M \in R(M_0)\) and \(M_p \in R(M_{p_0})\). If \(t' \in T\), by the liveness of \((N, M_0)\), \(\exists M' \in R(M)\), such that \(M(t)\). By the liveness of \((N, M_0)\) and \((\overline{N}_p, \overline{M}_{p_0})\), \(\exists M'' = [M_{(p,p)}, M_p] \in R(M')\), such that \(M''(t')\) where \(\overline{M} \in R(M)\), \(\overline{M}_p \in R(M_p)\). Then in \((N', M_0')\), \(t'\) is live. If \(t' \in T_p\), by the liveness of \((\overline{N}_p, \overline{M}_{p_0})\), \(\exists M_p \in R(M_p)\), such that \(M_p(t')\). By the liveness of \((N, M_0)\) and \((\overline{N}_p, \overline{M}_{p_0})\), \(\exists M''' = [M'_{(p,p)}, M_p] \in R(M')\), such that \(M'''(t')\), where \(\overline{M} \in R(M)\), \(\overline{M}_p \in R(M_p)\). Then in \((N', M_0')\), \(t'\) is live.

**Theorem 3.4** Suppose that \((N, M_0)\) is obtained from \((N', M_0')\) by \(p\)-subnet reduction operation. If \((N, M_0)\) is reversible and \(p \in p(p \in P) \land (M'_0(p) > 0)\), then \((N, M_0)\) and \((\overline{N}_p, \overline{M}_{p_0})\) are reversible.

**Proof.** Suppose that \((N, M_0)\) is not reversible, then \(\exists M_1 \in R(M_0)\), such that \(M_0 \notin R(M_1)\). By Supposition 3.1 and Definition 3.1-3.4, \(\exists M'_1 = [M_{(p,p)}, M_p]\), such that \(M_0' \notin R(M_1)\). This contradicts with the fact that \((N', M_0')\) is reversible. So, \(\exists M'', M''' \in R(M_0')\), such that \((N, M_0)\) (obtained from \((N', M_0'')\) ) and \((\overline{N}_p, \overline{M}_{p_0})\) (obtained from \((N', M_0'''\) ) are reversible. Since \(p_i \in p(p \in P) \land (M'_0(p) > 0)\), by Supposition 3.1 and Definition 3.1-3.4,
(N,M_0) (obtained from (N',M'_0)) and (\(\overline{N}_p, \overline{M}_{p_0}\)) (obtained from (N',M'_0)) are reversible.

**Theorem 3.5** Suppose that (N,M_0) is obtained from (N',M'_0) by p-subnet reduction operation. If (N,M_0) and (\(\overline{N}_p, \overline{M}_{p_0}\)) are reversible, then (N',M'_0) is reversible.

**Proof.** \(\forall M' \in R(M_0'), M'_0 = [M_{(p,p_0)}, M_{p_0}]\). Since (N,M_0) is reversible, then \(\forall M \in R(M_0), M_0 \in R(M)\). Since (\(\overline{N}_p, \overline{M}_{p_0}\)) is reversible, then \(\forall M_p \in R(M_{p_0})\), \(M_{p_0} \in R(M_p)\). By Supposition 3.1 and Definition 3.1-3.4, \(M'_0 \in R(M')\), i.e. (N',M'_0) is reversible.

IV. t-SUBNET REDUCTION METHOD

In this section we present t-subnet reduction operation. This operation preserves boundedness, liveness and reversibility.

In order to design and verify manufacturing system, we propose another subnet, named t-subnet.

*Fig.4.1. A t-subnet*

**Definition 4.1** A net \(N_0 = (P_0, T_0; F_0, W_0)\) is said to be a t-subnet of \(N = (P, T; F, W)\) iff,

1. \(*P_0 \cup P \subseteq T_0*\),
2. \(N_0\) is connected, \(\{t_i, t_o\} \subseteq T_0\) and \(t_i\) is the only input transition of \(N_0\), \(t_o\) is the only output transition of \(N_0\).

**Supposition 4.1** A t-subnet system \((N_0, M_{i0})\) contains t-subnet \(N_0\) and initial marking \(M_{i0}\), satisfy

1. In a process (tokens from outside flow into \(t_i\), pass \(N_0\) and then flow out from \(t_o\), \(t_o\) is fired, iff \(t_i\) is fired.
2. Before \(t_i\) is fired and after \(t_o\) is fired, \(\forall t \in T_0 - \{t_i, t_o\}, t\) can not be enabled.

(3) If \(P_0\) dose not contain token in initial state, \(P_0\) dose not contain token after a process; If \(P_0\) contains token(s) in initial state, the token(s) will come back to the initial state after a process.

**Definition 4.2** t-subnet reduction operation: a reduced net \(N' = (P', T'; F', W')\) is obtained from original Petri net \(N = (P, T; F, W)\) by using \(\tilde{t}\) to replace a t-subnet \(N_t = (P_t, T_t; F_t, W_t)\), where

1. \(P' = P - P_t;\)
2. \(T' = T \cup \{\tilde{t}\} - T_t;\)
3. \(F' = F \cup \{(p, \tilde{t}) | p \in t_o^*\} \cup \{(\tilde{t}, p) | p \in t_o^*\}
   - \{(p, t_i) | p \in t_i^*\} = \{(t_o, p) | p \in t_o^*\} - F_t\).

**Definition 4.3** \((N', M'_0)\) obtained from \((N, M_0)\) by t-subnet reduction operation comprises net \(N'\) and marking \(M'_0\), where \(M'_0 = M_{(P,P_t)}\) (where \(M_{(P,P_t)}\) is obtained from \(M\) by deleted the vector corresponding to \(P_t)\).

**Definition 4.4** A net \((\overline{N}_t, \overline{M}_{t0})\) is said to be a t-closed net if we add a transition \(t_i\) and arcs \(\{(p, t_i) | p \in t_o^*\} \cup \{(t_i, p) | p \in t_i^*\}\) to t-subnet \((N_t, M_{t0})\), and the marking of \((N_t, M_{t0})\) is preserved.

Note that in this section, let

- \((N', M'_0)\): the original net;
- \(N_t = (P_t, T_t; F_t, W_t)\): the p-subnet;
- \((N_i, M_{i0})\): the p-subnet system;
- \((\overline{N}_i, \overline{M}_{t0})\): the closed t-subnet system;
- \((N, M_0)\): the reduced net.

**Theorem 4.1** Suppose that \((N, M_0)\) is obtained from \((N', M'_0)\) by t-subnet reduction operation. Then \((N', M'_0)\) is bounded iff \((N, M_0)\) and \((\overline{N}_t, \overline{M}_{t0})\) are bounded.

**Proof.** (1) Since \((N, M_0)\) is bounded, then \(\forall p \in P, \exists k_1 > 0\) such that \(\forall M \in R(M_0), M(p) \leq k_1\).

Since \((\overline{N}_t, \overline{M}_{t0})\) is bounded, then \(\forall p \in P_t, \exists k_2 > 0\) such that \(\forall M_t \in R(M_{t0}), M_t(p) \leq k_2\).

Let \(k = k_1 + k_2, \forall p \in P, \forall M' \in R(M'_0)\) such that \(M'(p) \leq k\). So, \((N', M'_0)\) is bounded.

(2) Suppose that \((N, M_0)\) is unbounded, then \(\exists p \in P, \forall k > 0, \exists M \in R(M_0)\) such that...
$M(p) > k$. That is $\forall k > 0$, $\exists M' \in R(M_o')$ such that $M'(p) > k$. This contradicts with the fact that $(N', M_o')$ is bounded.

**Theorem 4.2** Suppose that $(N, M_o)$ is obtained from $(N', M_o')$ by t-subnet reduction operation. If $(N', M_o')$ is live and $t_i \subseteq \{p | p \in P \land (M'(p) > 0)\}$, then $(N, M_o)$ and $(\overline{N}, \overline{M}_o)$ are live.

**Proof.** Suppose $\Sigma = (N', M_o')$ (obtained from $\Sigma' = (N', M_o')$) is not live, i.e., $\exists M' \in R(M_o), \exists \overline{M} \in R(M)$ such that $\neg (\overline{M}[t_0])$. $\forall M'' \in R(M_o')$, since $M_o$ is the projection of $M_o'$ on $\Sigma$, then $M_o[\sigma > M[\overline{\sigma} > \overline{M}, \sigma, \overline{\sigma} \in T'']$. Let $t, \sigma, t_0$ replace $\overline{t}$ of $\sigma, \overline{\sigma}$, then we get $\sigma', \overline{\sigma''} \in T''$. By the liveness of $\Sigma'$, $M_o[\sigma' > M'[\overline{\sigma'} > \overline{M}']$. Since $M$ is the projection of $M'$ on $\Sigma$, and $\overline{M}$ is the projection of $\overline{M}'$ on $\Sigma$, according to $M, \exists M'' \in R(M_o'')$, $\exists \overline{M''} \in R(M)$ (if $t \in T - \{t_i\}$, then $t'_t = t$; if $t = t'$, then $t' = t_0$). According to $\forall \overline{M} \in R(M)$, there exists $\forall \overline{M''} \in R(M'')$. Since $\neg (\overline{M}[t_0])$, then $\exists M''[\overline{t'}]$, i.e., $\Sigma'$ is not live. This contradicts with the fact that $\Sigma'$ is live. Then $\exists M_0'', M_0'' \in R(M_0')$, such that $(N, M_0)$ (obtained from $(N', M_0'')$) and $(\overline{N}, \overline{M}_0)$ (obtained from $(N', \overline{M}_0'')$) are live. Since $t_i \subseteq \{p | p \in P \land (M(p) > 0)\}$, then $(N, M_0)$ (obtained from $(N', M_0'')$) and $(\overline{N}, \overline{M}_0)$ (obtained from $(N', \overline{M}_0'')$) are live.

**Theorem 4.3** Suppose that $(N, M_o)$ is obtained from $(N', M_o')$ by t-subnet reduction operation. If $(N, M_o)$ and $(\overline{N}, \overline{M}_o)$ are live, then $(N', M_o')$ is live.

**Proof.** $\forall t' \in T$, then $t' \in T - \{\overline{t}_i\}$ or $t' \in T - \{t_i, t_0\}$ or $t' \in \{t_i, t_0\}$. $\forall M' \in R(M_0')$, let $M = [M, M]$, by Supposition 4.1, $M \in R(M_o)$ and $t_i \in R(M_o)$.

1. If $t' \in T - \{\overline{t}_i\}$, by the liveness of $\Sigma = (N, M_o)$, $\exists \overline{M} \in R(M)$, such that $\overline{M}[t']$. Then $M[\sigma_o \supset M[\sigma > \overline{M}[t']$, where $\sigma_o$ and $\sigma$ are firing transition sequences of $\Sigma$.

(1.1) If $\overline{t} \notin \{t_i, t_0\}$, then $\exists M_o'[\sigma_o > \overline{M}[t']$, where $\sigma_o$ is a live transition sequence of $\Sigma$. $\overline{M}_o \in R(M_o)$, $\overline{M}_o' \in R(M_o')$, and $M_o[\sigma_o > M'[\sigma > \overline{M}[t']$, i.e., $t'$ is live in $(N', M_o')$.

(1.2) If $\overline{t} \in \{t_i, t_0\}$, let $M_0[\sigma_o > M[\sigma_i, \sigma_o \supset \overline{M}[t']$, by the liveness of $(N, M_0)$, $M[\sigma_o \supset M[\sigma_i, \sigma_o \supset \overline{M}[t']$ (where $\sigma_i$ is steps of $T_i$). So $t'$ is live in $\Sigma'$.

(2.1) If $t'$ dose not belong to the transition set of $\sigma_{i0}$, by the liveness of $\Sigma_i = (N, M_0)$, then $\exists M_i \in R(M_i)$, such that $M_0[\sigma_{i0} > M_i[\sigma_i > \overline{M}[t']$. Since $\Sigma = (N, M_0)$ is live, then $\exists M_0, M_0'[\sigma_o > M_i[\sigma_i > \overline{M}[t']$. \[\overline{M} \supset M'\] , then $M_0[\sigma_o \supset M'[\sigma_o > \overline{M}[t']$. Obviously, $t'$ is live in $\Sigma'$.

(2.2) If $t'$ belongs to the transition set of $\sigma_{i_0}$, then $[M_0, M_0']$ and $[\sigma_d i > \{M_i, M_i\}$ have the same meaning. Since $\Sigma$ is live, then $M_0[\sigma_o > M_i[\sigma_i > \overline{M}[t']$. Since $\Sigma = (N, M_0)$ is live, then $\exists M_0, M_0'[\sigma_o > M_i[\sigma_i > \overline{M}[t']$. Obviously, $t'$ is live in $\Sigma'$.
(3.2) If \( M_i \neq \emptyset \), by the liveness of \( \sum_i \), \( \exists \sigma_i \), such that \( M_i[\sigma_i] > M_i \). By the liveness of \( \sum \), \( \exists \sigma_0 \) such that \( M_0[\sigma_0] > M_0 \). Since \( \ast \tilde{t} = t' \), then in \( \sum' \), \( M_0''[\sigma_0'] > M'[\sigma,t,\sigma_0 > M'[t'] \). So \( t' \) is also live in \( \sum' \).

Since \( \forall t' \in T' \), \( t' \) is live, then \( \sum \) is live.

**Theorem 4.4** Suppose that \((N,M_0)\) is obtained from \((N',M_0')\) by t-subnet reduction operation. If \((N',M_0')\) is reversible and \( \ast \tilde{t} \subseteq \{p | p \in P \wedge (M'(p) > 0)\} \), then \((N,M_0)\) and \((\sum,\overline{M_0})\) are reversible.

**Proof.** Suppose \((N,M_0)\) is not reversible, then \( \exists M'_1 \in R(M_0) \), such that \( M_0 \notin R(M_1) \). Then \( \exists M'_1 = [M,M_1] \), such that \( M'_0 \notin R(M'_1) \). This contradicts with the fact that \((N',M_0')\) is reversible. So, \( \exists M''_0, M'''_0 \in R(M'_0) \), such that \((N,M_0)\) (obtained from \((N',M_0')\)) and \((\sum,\overline{M_0})\) (obtained form \((N',M_0')\)) are reversible. Since \( \ast \tilde{t} \subseteq \{p | p \in P \wedge (M'(p) > 0)\} \), then \((N,M_0)\) (obtained from \((N',M_0')\)) and \((\sum,\overline{M_0})\) (obtained form \((N',M_0')\)) are reversible.

**Theorem 4.5** Suppose that \((N,M_0)\) is obtained from \((N',M_0')\) by t-subnet reduction operation. If \((N,M_0)\) and \((\sum,\overline{M_0})\) are reversible, then \((N',M_0')\) is reversible.

**Proof.** Since \((N,M_0)\) is reversible, then \( \forall M \in R(M_0) \), \( M_0 \in R(M) \). Since \((\sum,\overline{M_0})\) is reversible, then \( \forall M_i \in R(M_0) \), \( M_0 \in R(M_i) \).

\( \forall M' \in R(M'_0) \), where \( M'_0 = [M_0,M'_0] \) and \( M'' = [M,M_1] \). Obviously, \( M'_0 \in R(M') \). So \((N',M_0')\) is reversible.

**V. APPLICATIONS**

In this section we apply results of Section 3 and Section 4 to reduce a flexible manufacturing system.

The manufacturing system consists of one workstation \( WS \) for assembly work and two machining centers for machining. Machining center_1 and \( WS \) share robot \( R_1 \). Machining center_2 and \( WS \) share robot \( R_2 \). The system runs as follows:

In the machining center_1, the intermediate parts are machined by machine \( M_1 \). Each part is fixtured to a pallet and loaded into the machine \( M_1 \) by robot \( R_1 \). After processing, robot \( R_1 \) unloads the final product, defixtures it and returns the fixture to \( M_1 \).

In the machining center_2, parts are machined first by machine \( M_2 \) and then by machine \( M_4 \). Each part is automatically fixtured to a pallet and loaded into the machine. After processing, robot \( R_2 \) unloads the intermediate part from \( M_4 \) into buffer B. At machining station \( M_4 \), intermediate parts are automatically loaded into \( M_4 \) and processed. When \( M_4 \) finishes processing a part, the robot \( R_2 \) unloads the final product, defixtures it and returns the fixture to \( M_4 \).

When workstation \( WS \) is ready to execute the assembly task, it requests robot \( R_1 \), robot \( R_2 \) and machine \( M_2 \) and acquires them if they are available. When the workstation starts an assembly task, it cannot be interrupted until it is completed. When \( WS \) completes, it releases the robot \( R_1 \) and robot \( R_2 \).

For the specification of the manufacturing system with Petri nets, each operation process is abstracted to a single place and each transition represents the start or/and completion of a process. Firstly, we give the Petri-net based model of the manufacturing system. Secondly, a reduced net system is obtained by p-subnet reduction method and t-subnet reduction method. Thirdly, we will analyze property preservation of the reduced net system.

The Petri-net based model \((N,M_0)\) of the original manufacturing system is illustrated in Fig.5.1.

The meaning of places and transitions of Fig.5.1 is as below.

\( p_{11} \): The pallet is available and semi-finished parts are available; \( p_{12} \): Apply for the use of robot \( R_1 \); \( p_{13} \): Obtain the right to use \( R_1 \), and \( R_1 \) fix the pallet on machine \( M_1 \); \( p_{14} \): Parts on machine \( M_1 \) are reprocessed; \( p_{15} \): Machine \( M_1 \) is available; \( p_{16} \): Apply for the use of robot \( R_1 \); \( p_{17} \): Unload the finished product and return the pallet; \( p_{21} \): The pallet is available and the original parts are available; \( p_{22} \): Machine \( M_3 \) process parts on pallet; \( p_{23} \): Machine \( M_3 \) is available; \( p_{24} \): Prepare buffer intermediate parts; \( p_{25} \): Robot \( R_2 \) put intermediate parts into buffer \( B \); \( p_{26} \): Buffer \( B \) is available; \( p_{27} \): Intermediate parts are available; \( p_{28} \): Intermediate parts
are processed on machine $M_4$, and robot $R_2$ unload finished product on $M_4$ and return the pallet; $p_{29}$: Machine $M_4$ is available; $p_{31}$: Machine $M_2$ is available; $p_{32}$: Apply for the use of robot $R_1$ and robot $R_2$; $p_{33}$: WS obtain the right to use robot $R_1$ and robot $R_2$; $p_{34}$: Execute the first assembly in $WS$; $p_{35}$: Execute the last assembly in $WS$; $p_{r11}$: Robot $R_1$ is available; $p_{r12}$: Robot $R_1$ is cleaned and lubricated; $p_{r13}$: Replace related accessories of $R_1$; $p_{r14}$: Robot $R_1$ is placed on standby for working; $p_{r21}$: Robot $R_2$ is available; $p_{r22}$: Robot $R_2$ is cleaned and lubricated; $p_{r23}$: Replace related accessories of $R_2$; $p_{r24}$: Robot $R_2$ is placed on standby for working.

Fig. 5.1 The original net system

$t_{11}$: Execute $p_{13}$; $t_{12}$: Finish $p_{13}$ and execute $p_{14}$; $t_{13}$: Finish $p_{14}$ and execute $p_{17}$; $t_{14}$: Finish $p_{17}$; $t_{21}$: Execute $p_{22}$; $t_{22}$: Finish $p_{22}$ and execute $p_{24}$; $t_{23}$: Finish $p_{24}$ and execute $p_{25}$; $t_{24}$: Finish $p_{25}$ and execute $p_{27}$; $t_{25}$: Finish $p_{27}$ and execute $p_{28}$; $t_{26}$: Finish $p_{28}$; $t_{31}$: Start to obtain the right to use robot $R_1$ and $R_2$; $t_{32}$: Start the first assembly in $WS$; $t_{33}$: Start the last assembly in $WS$; $t_{34}$: Finish assemblies in $WS$; $t_{r11}$: Start to clean and lubricate robot $R_1$; $t_{r12}$: Start to replace related accessories of $R_1$; $t_{r13}$: Finish cleaning and lubricating work of robot $R_1$; $t_{r14}$: Finish replacing related accessories of robot $R_1$; $t_{r21}$: Start to clean, lubricate and replace related accessories of $R_2$; $t_{r22}$: Finish cleaning, lubricating and replacing related accessories of robot $R_2$. The reduction process consists of the following three steps.

Step 1: $(N_1, M_1)$ (Fig. 5.2) is obtained by p-subnet reduction method. p-subnet $(N_{p1}, M_{p10})$ (generated by $\{p_{r11}, p_{r12}, p_{r13}, p_{r14}, t_{r11}, t_{r12}, t_{r13}, t_{r14}\}$) is reduced to $p_{r1}$. p-subnet $(N_{p2}, M_{p20})$ (generated by $\{p_{r21}, p_{r22}, p_{r23}, p_{r24}, t_{r21}, t_{r22}\}$) is reduced to $p_{r2}$.

By Theorem 3.1-3.5, $(N, M_0)$ is bounded, live and reversible, iff $(N_1, M_1)$ is bounded, live and reversible.

Fig. 5.2 The reduced net $(N_1, M_1)$

Step 2: $(N_2, M_2)$ (Fig. 5.3) is obtained by t-subnet reduction method. t-subnet $(N_{t1}, M_{t10})$ (generated by $\{p_{12}, p_{13}, t_{11}, t_{12}\}$) is reduced to $t_{1}'$. t-subnet $(N_{t2}, M_{t20})$ (generated by $\{p_{16}, p_{17}, t_{13}, t_{14}\}$) is reduced to $t_{2}'$. t-subnet $(N_{t3}, M_{t30})$ (generated by $\{p_{31}, p_{32}, p_{33}, p_{34}, p_{35}, t_{31}, t_{32}, t_{33}, t_{34}\}$) is reduced to $t_{3}'$. t-subnet $(N_{t4}, M_{t40})$ (generated by $\{p_{22}, p_{23}, t_{21}, t_{22}\}$) is reduced to $t_{4}'$. t-subnet $(N_{t5}, M_{t50})$ (generated by $\{p_{25}, p_{26}, t_{23}, t_{24}\}$) is reduced to $t_{5}'$. t-subnet $(N_{t6}, M_{t60})$ (generated by $\{p_{28}, p_{29}, t_{25}, t_{26}\}$) is reduced to $t_{6}'$. By Theorem 4.1-4.5, $(N_1, M_1)$ is bounded, live and reversible, iff $(N_2, M_2)$ is bounded, live and reversible.

Fig. 5.3 The reduced net $(N_2, M_2)$
Step 3: \((N_3, M_3)\) (Fig. 5.4) is obtained by deleted places \(p_{r1}\) and \(p_{r2}\). Since \(p_{r1} = p_{r1}^*\), \(M_2(p_{r1}) > 0\) and \(p_{r2} = p_{r2}^*\), \(M_2(p_{r2}) > 0\), then \((N_2, M_2)\) is bounded, live and reversible, iff \((N_3, M_3)\) is bounded, live and reversible.

It is easy to see that \((N_3, M_3)\) is bounded, live and reversible (5)). By Theorem 3.1-3.5 and Theorem 4.1-4.5, the original net system \((N, M_0)\) is bounded, live and reversible.

VI. CONCLUSIONS

In this paper we investigate dynamic property preservations of Petri reduction net. Two Petri net reduction methods are proposed, which are the key methods to ensure the reduced net preserving well behaved properties. Conditions of boundedness, liveness and reversibility preservations of ordinary Petri reduction net are proposed. As a consequence, this result can be applied nicely to solve some of the subsystem reducing problem in manufacturing engineering, software engineering and management engineering. Further research is needed to give more general conditions to investigate other property preservations of the reduction net.

ACKNOWLEDGMENT

This work was financially supported by the National Natural Science Foundation of China under Grant No. 60573012, 60672180 and 60721061, the National Grand Fundamental Research 973 Program of China under Grant No. 2002cb312200, and CAS Key Laboratory of Computer Sciences (SYSKF0903).

REFERENCES


Chuanliang Xia received his Ph. D. degree in Computer Science from Academy of Mathematics and Systems Science, Chinese Academy of Science in 2006. He was a postdoctor of State Key Laboratory for Computer Science, Institute of Software, Chinese Academy of Science. He is currently an associate professor of School of Computer Science and Technology, Shandong Jianzhu University. His current research interests include Petri nets, algorithm design and analysis, model checking, parallel computing etc.