Petri 网化简操作及其在系统验证中的应用*)

夏传良[1,2] 徐 进[1] 张光卫[3]
(山东建筑工程学院计算机科学与技术系 济南 250014)[1]
(中国科学院数学与系统科学研究院计算机科学研究室 北京 100080)[2]
(北京航空航天大学软件开发环境国家重点实验室 北京 100083)[3]

摘 要 针对柔性制造系统的验证问题,提出了用 Petri 网化简操作解决问题的方案。给出了两种化简操作。研究了这两种化简操作的动态性质保持问题,给出了化简后的 Petri 网保持活性、有界性、可回复性的一组充分条件或充要条件。对一个柔性制造系统进行了验证。本文的结果可为 Petri 网系统静态和动态性质的考察提供有效途径,为复杂大系统的分析提供重要手段,并特别适合于柔性制造系统的验证,具有一定的实用价值。

关键词 Petri 网,化简操作,活性,有界性,系统验证,柔性制造系统

The Petri Net Reduction and its Application in System Verification

XIA Chuan-Liang[1,2] XU Jin[1] ZHANG Guang-Wei[3]
(Department of Computer Science and Technology, Shandong Institute of Architecture and Engineering, Jinan 250014)[1]
(Department of Computer Science, Academy of Mathematics and System Sciences, The Chinese Academy of Sciences, Beijing 100080)[2]
(State Key Lab. of Software Development Environment, Beijing University of Aeronautics and Astronautics, Beijing 100083)[3]

Abstract A scheme is obtained using some kinds of Petri net reduction, according to the verification of flexible manufacturing system. Two kinds of reductions are proposed and their dynamic property preservations have been investigated. A group of sufficient conditions or sufficient and necessary conditions of liveness preservation, boundedness preservation and reversibility preservation are presented. 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. The reduction method is especially fit for system verification and practical to use in reality.

Keywords Petri nets, Reduction, Liveness, Boundedness, System verification, Flexible manufacturing system

1 引论

在系统设计中经常遇到资源共享和系统化简(system reduction)问题。每个系统都需要一些资源,例如在柔性制造系统中,机器人、机器或生产线等都被称为资源;而在软件工程中,缓冲器、数据类型库、服务器和数据库等则作为计算性资源。有诸多原因使得这些资源在系统中被多个过程所共享。例如,在柔性制造系统中,为了减少昂贵的机器人、机器等的空闲时间,它们经常被多个过程所共享。注意,这里的“共享”并不意味着同时使用。在软件系统中用重入代码(re-enterable code)可达到同时使用的目的,但是在柔性制造系统中,“共享”资源只能被某个过程所独占,直到该过程结束才能释放,再被其它过程使用。在拥有多个共享资源的过程中,一个错误的占用和释放顺序往往可使系统产生死锁或溢出。因此,对一个系统的可行性进行验证是十分必要的。从系统验证的角度看,化简的目的是当把被抽象化的部分看作一个单独的具有某种功能的个体而忽略内部逻辑时,来检查系统是否正确。对系统的设计和验证需要一整套完整的理论和方法。Petri 网是一种系统的数学和图形的建模和分析工具,特别适用于对具有同步、并发、冲突的离散事件系统进行建模和分析。它广泛应用于复杂系统的设计与分析中,如计算机系统、分布式并行处理系统、柔性制造系统等。用 Petri 网来表示并发、互斥、同步显得直接、自然和精确,同时由于 Petri 网具有坚实的数学基础,因此它为系统模型的分析和验证提供了一种有效的方法。但是,当建模的系统大而且复杂时,就会由于状态空间爆炸而带来系统分析上的高复杂性。有一种重要的方法可用来降低大系统建模分析的复杂度,这就是系统的化简操作。化简操作就是把复杂的 Petri 网系统中的某些子网化简为库所或变迁,并在化简过程中保持原系统的某些重要性质,如有界性、活性和可回复性不变,从而降低建模和分析的复杂度。

Petri 网的化简思想一直为理论界和工程界所关注,已做了大量的工作。Petri 网化简操作的研究从一些简单的模式变更开始[1,2]。Desel[1]指出,一个活的并且安全的自由选择网(FC)可被化简为一个活的且安全的标识图(marked graph),或一个活的且安全的状态机(state machine);Esparza[2]提出了将一个活的、有界的 FC 网化简为一个只有一个库所和一个变迁的回路的若干规则;一个重要的结果是在合并了一个 FC 网[3~5]或 AC 网[6]的若干库所后得到的网保持良构(well-formedness)和 Commoner 性质;文[7]提出了对一个

*)国家自然科学基金(60073013,60473007)、国家重点基础研究发展规划 973 项目(G19980304016)和中国科学院管理、决策与信息系统开放实验室(MADIS)资助。**夏传良** 副教授,博士,主要研究领域为 Petri 网、算法设计与分析、计算机网络与通信;**徐 进** 副研究员,主要研究领域为计算机应用;**张光卫** 博士研究生,主要研究领域为复杂网络与网络智能、不确定性人工智能。

随机 Petri 网(SPN)进行抽象的方法，给定的 SPN 被转化为一个状态变迁系统(STS)，并且对 STS 进行了化简；为了避免出现状态空间爆炸问题，文[8]关于面向对象的 Petri 网(object-oriented Petri net)，给出了一种称为偏序化简的方法，该方法可应用于模型检测，并特别适用于处理软件系统；文[9]对于时间 Petri 网给出了 9 条化简规则，引入了一个等价关系，并证明了按照这些化简规则化简后的网与原网等价；文[10]给出了一种状态空间化简方法，应用于代数配置；文[11]对于变迁系统(TS)给出了一个新的验证算法，该算法用于解决有限状态模型检测问题，同时也给出了一种新的系统化简方法；……这些工作均针对 Petri 网化简操作和网性质的分析。

上述工作虽然给出了一些化简方法，但一般不适用于柔性制造系统的设计和验证。为了对柔性制造系统进行验证，本文根据共享资源或被替换部分的结构，给出了两种化简操作：P-型子网化简操作和 T-型子网化简操作。因为无死锁性和无溢出性是系统设计和分析中的两个十分重要的性质，它们分别对应着 Petri 网系统的活性和有界性，而可回复性保证了系统从任何状态都能回到初始状态，所以本文对 P-型子网和 T-型子网的化简的活性、有界性和可回复性的继承关系进行了比较深入的研究。给出了一组化简后的目标网保持活性、有界性和可回复性的充分条件或充要条件。最后用这两种操作验证了一个柔性制造系统，证实了本文结果的适用性。

本文结构如下：第 2 节给出相关的基本概念、符号；第 3 节提出 P-型子网化简操作，研究它对有界性、活性和可回复性的保持性问题；第 4 节提出 T-型子网化简操作，研究它对有界性、活性和可回复性的保持性问题；第 5 节对一个柔性制造系统进行验证；最后总结全文。

2 基本定义和符号

关于 Petri 网的基本概念和术语可参见文[12～14]，这里只引入与本文相关的少数几个概念。

定义 2.1 设 $N=(P,T;F,W)$ 是一个 Petri 网，$\Sigma=(N,M_0)$ 是一个 Petri 网系统，$M\in R(M_0)$，

(1) 变迁 $t\in T$ 称为在 M 下使能，当且仅当对 $\forall p\in {}^{\bullet}t$：$M(p)\geqslant W(p,t)$，记作 $M[t>$；

(2)若 $M[t_1>M_1[t_2>\cdots M_{n-1}[t_n>M_n$（其中 $M_i\in R(M_0)$，$t_i\in T$，$i=1,2,\cdots,n$），则称 $\sigma=t_1t_2\cdots t_n$ 为 $\Sigma=(N,M_0)$ 的一个可引发变迁序列，记作 $M[\sigma>M_n$。

定义 2.2 设 $\Sigma=(N,M_0)$ 是一个 Petri 网系统。

(1) 变迁 $t\in T$ 是活的，当且仅当 $\forall M\in R(M_0)$，$\exists M'\in R(M)$，$M'[t>$；

(2) Σ 是活的，当且仅当 $\forall t\in T$，t 是活的。

定义 2.3 设 $\Sigma=(N,M_0)$ 是一个 Petri 网系统。

(1) 库所 $p\in P$ 是有界的，当且仅当存在常数 $k>0$，使得 $M(p)\leqslant k$，$\forall M\in R(M_0)$；

(2) Σ 是有界的，当且仅当 $\forall p\in P$，p 是有界的。

定义 2.4 设 $\Sigma=(N,M_0)$ 是一个 Petri 网系统。Σ 是可回复的，当且仅当 $M_0\in R(M)$，$\forall M\in R(M_0)$。

3 P-型子网化简操作

针对柔性制造系统的设计和验证，提出了 P-型子网，简单示例如图 1。

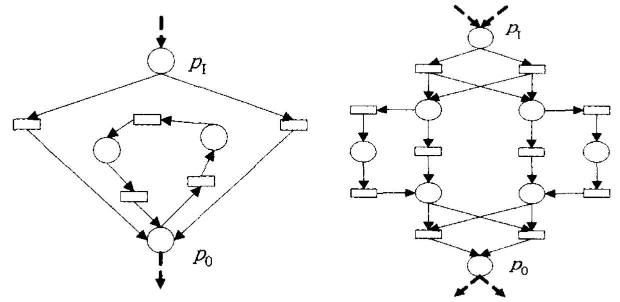

图 1 P-型子网简单示例

定义 3.1 设 $N=(P,T;F,W)$ 是一个 Petri 网，$N_0=(P_0,T_0;F_0,W_0)$ 是 N 的一个子网，若满足：

(1) ${}^{\bullet}T_0\cup T_0^{\bullet}\subseteq P_0$；

(2) N_0 是连通的，并且 $\{p_I,p_O\}\subseteq P_0$，其中 p_I 是唯一的输入库所，p_O 是唯一的输出库所；则称 N_0 为 N 的一个 P-型子网。

假定 3.1 P-型子网系统 (N_0,M_{P0}) 由 P-型子网 N_0 和初始标识 M_{P0} 构成，并且满足：

(1)初始标识(托肯)只能出现在 p_I 中；

(2)在 (N_0,M_{P0}) 的一次执行过程(从托肯流入 p_I 至由 p_O 流出)中，从外部流入 p_I 的托肯数与流出 p_O 的托肯数相等，并且一次执行过程结束后，P_0 中的每个库所都不含托肯。

定义 3.2 P-型子网化简操作：将 Petri 网 $N=(P,T;F,W)$ 中的 P-型子网 $N_P=(P_P,T_P;F_P,W_P)$ 化简为一个库所 $\tilde{p}$（即用 $\tilde{p}$ 来替换 $N_P=(P_P,T_P;F_P,W_P)$），得到 Petri 网 $N'=(P',T';F',W')$，其中

(1) $P'=P\cup\{\tilde{p}\}-P_P$；(2) $T'=T-T_P$；

(3) $F'=F\cup\{(t,\tilde{p})\mid t\in {}^{\bullet}p_I\}\cup\{(\tilde{p},t)\mid t\in p_O{}^{\bullet}\}-\{(t,p_I)\mid t\in {}^{\bullet}p_I\}-\{(p_O,t)\mid t\in p_O{}^{\bullet}\}-F_P$。

定义 3.3 经 P-型子网化简操作得到的网系统为 $\Sigma'=(N',M'_0)$，其中

$$M'_0=\begin{cases}[M_{(P\backslash P_P)0},0] & M_0(p_I)=0\\ [M_{(P\backslash P_P)0},n] & M_0(p_I)=n\quad(n\geqslant 1)\end{cases}$$

其中 $M_{(P\backslash P_P)}$ 为 M 中去掉 P_P 所对应的分量以后的向量。

定义 3.4 经 P-型子网化简操作化简掉的 P-型子网系统为 (N_P,M_{P0})，其中，M_{P0} 为 M_0 在 P_P 上的投影。

定义 3.5 P-型闭网系统：为 P-型子网系统 (N_0,M_{P0}) 增加一个变迁 t_P 和两条有向弧 (p_O,t_P)，(t_P,p_I)，并且标识不变，得到 P-型闭网系统 $(\overline{N}_P,\overline{M}_{P0})$。

定义 3.6 P-型子网恢复操作：将 Petri 网 $N=(P,T;F,W)$ 中的库所 $\tilde{p}$ 恢复为 P-型子网 $N_P=(P_P,T_P;F_P,W_P)$（即用 $N_P=(P_P,T_P;F_P,W_P)$ 来替换 $\tilde{p}$），得到 Petri 网 $N'=(P',T';F',W')$，其中

(1) $P'=(P-\{\tilde{p}\})\cup P_P$；(2) $T'=T\cup T_P$；

(3) $F'=F\cup\{(t,p_I)\mid t\in {}^{\bullet}\tilde{p}\}\cup F_P\cup\{(p_O,t)\mid t\in\tilde{p}^{\bullet}\}-\{(t,\tilde{p})\mid t\in {}^{\bullet}\tilde{p}\}-\{(\tilde{p},t)\mid t\in\tilde{p}^{\bullet}\}$。

定义 3.7 经 P-型子网恢复操作后得到的网系统 (N',M'_0) 由恢复操作后得到的网 N' 和初始标识 M'_0 构成，其中

$$M'_0=\begin{cases}[M_{(P\backslash\tilde{p})0},\theta_P] & M_0(\tilde{p})=0,\\ [M_{(P\backslash\tilde{p})0},M_{P0}] & M_0(\tilde{p})>0。\end{cases}$$

其中 $M_{(P\backslash\tilde{p})}$ 为 M 中去掉 $\tilde{p}$ 所对应的分量以后的向量，θ_P 是 M_P 的零向量。

注：为避免引起混乱，在以下定理 3.1～3.5 中，用(N',M'_0)表示原网系统，$N_P=(P_P,T_P;F_P,W_P)$为它的 P-型子网；(N,M_0)为化简后得到的网系统；(N_P,M_{P0})为化简掉的 P-型子网系统，其对应的 P-型闭网系统为$(\bar{N}_P,\bar{M}_{P0})$。

定理 3.1 设(N,M_0)是由(N',M'_0)经 P-型子网化简操作得到的 Petri 网系统，则(N',M'_0)是有界的当且仅当(N,M_0)与$(\bar{N}_P,\bar{M}_{P0})$都是有界的。

证明：(1)先证充分性。因为(N,M_0)是有界的，则$\forall p\in P$，存在正常数k_1使得$M(p)\leqslant k_1$，$\forall M\in R(M_0)$。显然，$\forall p\in P-\{\tilde{p}\}$，$M_{(P\backslash\tilde{p})}(p)\leqslant k_1$(其中$M_{(P\backslash\tilde{p})}$为$M$中去掉$\tilde{p}$所对应的分量以后的向量)。因为$(\bar{N}_P,\bar{M}_{P0})$有界，则$\forall p\in P_P$，存在一个正常数$k_2$使得$M_P(p)\leqslant k_2\ \forall M_P\in R(M_{P0})$。令$k=k_1+k_2$，根据假定 3.1 和定义 3.6、3.7 知，$\forall p\in P'$，$M'(p)=[M_{P\backslash\tilde{p}},M_P](p)\leqslant k\ \forall M'\in R(M'_0)$，所以$(N',M'_0)$有界。

(2)再证必要性。采用反证法，假设(N,M_0)无界，则$\exists p\in P$，$\forall k>0$，$\exists M\in R(M_0)$且$M(p)>k$。根据假定 3.1 和定义 3.2～3.7，$\forall k>0$，$\exists M'\in R(M'_0)$且$M'(p)>k$。这与题设(N',M'_0)有界矛盾。

定理 3.2 设(N,M_0)是由(N',M'_0)经 P-型子网化简操作得到的 Petri 网系统，如果(N',M'_0)是活的，并且$p_I\in\{p\mid(p\in P')\wedge(M'_0(p)>0)\}$，那么$(N,M_0)$与$(\bar{N}_P,\bar{M}_{P0})$都是活的。

证明：采用反证法。假设$\forall M_0''\in R(M'_0)$，由(N',M_0'')中经化简操作得到的(N,M_0)不活，亦即$\exists M\in R(M_0)$，$\exists t\in T$，对$\forall\bar{M}\in R(M)$，都有$\neg(\bar{M}[t>)$。因为(N',M'_0)是活的，$M_{(P\backslash\tilde{p})0}$是$M_0''$在$P-\{\tilde{p}\}$上的投影，所以可记$M_{(P\backslash\tilde{p})0}[\sigma>M_{(P\backslash\tilde{p})}[\bar{\sigma}>\bar{M}_{(P\backslash\tilde{p})}$，$\sigma,\bar{\sigma}\in T$(其中由定义 1 知，$\sigma,\bar{\sigma}$为$(N,M_0)$上的可引发变迁序列)，现在加入$(N_P,M_{P0})$中的变迁(或变迁步)$\sigma_P$，得到$\sigma',\sigma''\in T''$。从$M_{(P\backslash\tilde{p})0}[\sigma>M_{(P\backslash\tilde{p})}[\bar{\sigma}>\bar{M}_{(P\backslash\tilde{p})}$，$\sigma,\bar{\sigma}\in T$，根据假定 3.1、定义 3.6、定义 3.7 和(N',M'_0)的活性易知，$M'_0[\sigma'>M''[\sigma''>\bar{M}'$，并且$M_{(P\backslash\tilde{p})}$是$M''$在$P-\{\tilde{p}\}$上的投影，$\bar{M}_{P\backslash\tilde{p}}$是$\bar{M}''$在$P-\{\tilde{p}\}$上的投影，这样对应于$M_{(P\backslash\tilde{p})}$，$\exists M''\in R(M_0'')$，$\exists t'\in T''$(其中$t'=t$)，使得对应于$\forall\bar{M}_{P\backslash p}$有$\forall\bar{M}''\in R(M'')$，由$\neg(\bar{M}_{(P\backslash p)}[t'>)$可推知$\neg(\bar{M}''[t'>)$，从而$(N',M'_0)$不活，矛盾。因此$\exists M_0'',M_0'''\in R(M'_0)$，使得从$(N',M_0'')$中经化简操作得到的$(N,M_0)$和从$(N',M_0''')$中经化简操作得到的$(\bar{N}_P,\bar{M}_{P0})$都是活的。又因为$p_I\in\{p\mid(p\in P')\wedge(M'_0(p)>0)\}$，所以从$(N',M'_0)$中经化简操作得到的$(N,M_0)$和$(\bar{N}_P,\bar{M}_{P0})$都是活的。

定理 3.3 设(N,M_0)是由(N',M'_0)经 P-型子网化简操作得到的 Petri 网系统，如果(N,M_0)与$(\bar{N}_P,\bar{M}_{P0})$都是活的，那么(N',M'_0)是活的。

证明：在(N',M'_0)中，$\forall t'\in T'$，则有$t'\in T$或者$t'\in T_P$。对$\forall M'\in R(M'_0)$令$M'=[M_{(P\backslash\tilde{p})},M_P]$，根据假定 3.1 有，$M\in R(M_0)$和$M_P\in R(M_{P0})$。如果$t'\in T$，由$(N,M_0)$的活性，对$M\in R(M_0)$，$\exists\bar{M}\in R(M)$，使得$\bar{M}[t'>$。根据$(N,M_0)$和$(\bar{N}_P,\bar{M}_{P0})$的活性以及假定 3.1、定义 3.6 和定义 3.7 知，$\exists\bar{M}'=[\bar{M}_{(P\backslash\tilde{p})},\bar{M}_P]\in R(M')$，使得$\bar{M}'[t'>$，其中，$\bar{M}\in R(M)$，$\bar{M}_P\in R(M_P)$。从而$t'$在$(N',M'_0)$中是活的。如果$t'\in T_P$，由$(\bar{N}_P,\bar{M}_{P0})$的活性，对$M_P\in R(M_{P0})$，$\exists\bar{M}_P\in R(M_P)$，使得$\bar{M}_P[t'>$。根据$(N,M_0)$和$(\bar{N}_P,\bar{M}_{P0})$的活性以及假定 3.1、定义 3.6 和定义 3.7 知，$\exists\bar{M}'=[\bar{M}'_{(P\backslash\tilde{p})},\bar{M}_P]\in R(M')$，使得$\bar{M}'[t'>$，其中，$\bar{M}'\in R(M)$，$\bar{M}_P\in R(M_P)$。于是$t'$在$(N',M'_0)$中是活的。所以由$t'$的任意性可知，$(N',M'_0)$是活的。

定理 3.4 设(N,M_0)是由(N',M'_0)经 P-型子网化简操作得到的 Petri 网系统，如果(N',M'_0)是可回复的，并且$p_I\in\{p\mid(p\in P')\wedge(M'_0(p)>0)\}$，那么$(N,M_0)$与$(\bar{N}_P,\bar{M}_{P0})$都是可回复的。

证明：采用反证法。假设(N,M_0)不是可回复的，则$\exists M_1\in R(M_0)$，使得$M_0\notin R(M_1)$，根据假定 3.1 和定义 3.2～3.7 知，$\exists M_1'=[M_{(P\backslash\tilde{p})1},M_{P1}]$，使得$M'_0\notin R(M_1')$，这与题设$(N',M'_0)$是可回复的相矛盾。因此，$\exists M_0'',M_0'''\in R(M'_0)$，使得从$(N',M_0'')$中得到的$(N,M_0)$和从$(N',M_0''')$中得到的$(\bar{N}_P,M_{P0})$都是可回复的。因为$p_I\in\{p\mid(p\in P')\wedge(M'_0(p)>0)\}$，所以根据假定 3.1 和定义 3.2～3.7 易知，从(N',M'_0)中得到的(N,M_0)和$(\bar{N}_P,\bar{M}_{P0})$都是可回复的。

定理 3.5 设(N,M_0)是由(N',M'_0)经 P-型子网化简操作得到的 Petri 网系统，如果(N,M_0)与$(\bar{N}_P,\bar{M}_{P0})$都是可回复的，那么(N',M'_0)是可回复的。

证明：$\forall M'\in R(M'_0)$，根据定义 3.6 和定义 3.7，$M'_0=[M_{(P\backslash\tilde{p})0},M_{P0}]$，$M'=[M_{(P\backslash\tilde{p})},M_P]$。因为$(N,M_0)$是可回复的，则$\forall M\in R(M_0)$，$M_0\in R(M)$。又因为$(\bar{N}_P,\bar{M}_{P0})$是可回复的，则$\forall M_P\in R(M_{P0})$，$M_{P0}\in R(M_P)$。从假定 3.1 和定义 3.2～3.7 易知，$M'_0\in R(M')$，即$(N',M'_0)$是可回复的。

4 T-型子网化简操作

针对柔性制造系统的设计和验证，提出了 T-型子网，简单示例如图 2。

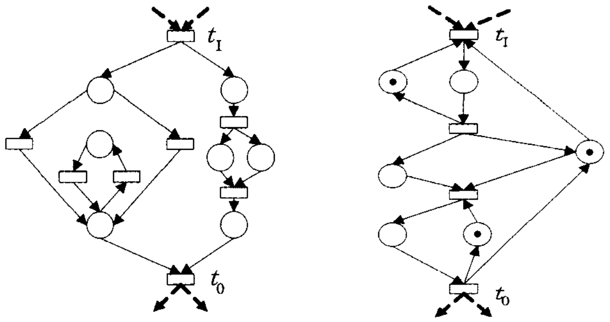

图 2 T-子网简单示例

定义 4.1 设$N=(P,T;F,W)$是一个 Petri 网，$N_0=(P_0,T_0;F_0,W_0)$是N的一个子网，若满足：

(1)$^{\bullet}P_0\cup P_0{}^{\bullet}\subseteq T_0$；

(2) N是连通的，并且$\{t_I,t_O\}\subseteq T_0$，t_I是唯一的输入变迁，t_O是唯一的输出变迁；则称N_0为N的一个 T-型子网。

假定 4.1 T-型子网系统(N_0,M_{T0})由 P-型子网N_0和初始标识N_{T0}构成，并且满足：在(N_0,M_{T0})的一次执行过程(从托肯流入t_I至由t_O流出)中，从外部流入p_I的托肯数与流出p_O的托肯数相等；若P_0中在初始状态下有托肯，一次执行过程结束后，P_0中所有托肯又恢复为原来状态。

定义 4.2 T-型子网化简操作：将 Petri 网$N=(P,T;F,W)$中的 T-型子网$N_T=(P_T,T_T;F_T,W_T)$化简为一个变迁$\tilde{t}$(即用$\tilde{t}$来替换$N_T=(P_T,T_T;F_T,W_T)$)，得到 Petri 网$N'=(P',T';F',W')$，其中

(1)$P'=P-P_T$；(2)$T'=T\cup\{\tilde{t}\}-T_T$；

(3)$F'=F\cup\{(p,\tilde{t})\mid p\in{}^{\bullet}t_I\}\cup\{(\tilde{t},p)\mid p\in t_O{}^{\bullet}\}-\{(p,t_I)\mid p\in{}^{\bullet}t_I\}-\{(t_O,p)\mid p\in t_O{}^{\bullet}\}-F_T$。

定义 4.3 经 T-型子网化简操作得到的网系统为(N',M'_0)，其中$M'_0=M_{(P\backslash P_T)0}$(其中$M_{(P\backslash P_T)}$为$M$中去掉$P_T$所对应的分量以后的向量)。

定义 4.4 经 T-型子网化简操作化简掉的 T-型子网系统为(N_T,M_{T0})，其中M_{T0}为M_0在P_T上的投影。

定义 4.5 在网系统(N,M_0)中取子网由$^{\cdot}t_I$开始经过N_T到$t_O{}^{\cdot}$，并增加变迁t_T以及弧集$\{(p,t_T)\mid p\in t_O{}^{\cdot}\}\cup\{(t_T,p)\mid p\in{}^{\cdot}t_I\}$，且对任意的$p\in{}^{\cdot}t_I$：$w(t_T,p)=w(p,t_I)$；对任意的$p\in t_O{}^{\cdot}$：$w(p,t_T)=w(t_O,p)$，并且标识不变，得到 T-型闭网系统$(\overline{N}_T,\overline{M}_{T0})$。

定义 4.6 T-型子网恢复操作：将 Petri 网$N=(P,T;F,W)$中的变迁$\tilde{t}$恢复为 T-型子网$N_T=(P_T,T_T;F_T,W_T)$(即用 T-型子网$N_T=(P_T,T_T;F_T,W_T)$来替换$\tilde{t}$)，得到 Petri 网$N'=(P',T';F',W')$，其中

(1)$P'=P\cup P_T$；(2)$T'=T\cup T_T-\{\tilde{t}\}$；

(3)$F=F\cup\{(p,t_I)\mid p\in{}^{\cdot}\tilde{t}\}\cup\{(t_O,p)\mid p\in\tilde{t}^{\cdot}\}-\{(p,\tilde{t})\mid p\in{}^{\cdot}\tilde{t}\}-\{(\tilde{t},p)\mid p\in\tilde{t}^{\cdot}\}$。

定义 4.7 经 T-型子网恢复操作后得到的网系统(N',M'_0)由恢复操作后得到的网N'和初始标识M'_0构成，记$M'_0=[M_0,M_{T0}]$，其中M_0,M_{T0}分别是N与N_T的初始标识。

注：为避免引起混乱，在以下定理 4.1～4.5 中，用(N',M'_0)表示原网系统，$N_T=(P_T,T_T;F_T,W_T)$为它的 T-型子网；(N,M_0)为化简后得到的网系统；(N_T,M_{T0})为化简掉的 T-型子网系统，其对应的 T-型闭网系统为$(\overline{N}_T,\overline{M}_{T0})$。

定理 4.1 设(N,M_0)是由(N',M'_0)经 T-型子网化简操作得到的 Petri 网系统，则(N',M'_0)是有界的当且仅当(N,M_0)与$(\overline{N}_T,\overline{M}_{T0})$都是有界的。

证明：(1)先证明充分性。因为(N,M_0)是有界的，则$\forall p\in P$，存在正常数k_1使得$M(p)\leqslant k_1\ \forall M\in R(M_0)$。又因为$(\overline{N}_T,\overline{M}_{T0})$有界，则$\forall p\in P_T$，存在一个正常数$k_2$使得$M_T(p)\leqslant k_2\ \forall M_T\in R(M_{T0})$。令$k=k_1+k_2$，根据定义 4.6，$\forall p\in P'$，$M'(p)\leqslant k\ \forall M'\in R(M'_0)$，所以$(N',M'_0)$有界。

(2)再证明必要性。采用反证法，假设(N,M_0)无界，则$\exists p\in P,\forall k>0,\exists M\in R(M_0)$且$M(p)>k$。根据定义 4.6，$\forall k>0,\exists M'\in R(M'_0)$且$M'(p)>k$。这与题设$(N',M'_0)$有界相矛盾。

定理 4.2 设(N,M_0)是由(N',M'_0)经 T-型子网化简操作得到的 Petri 网系统，如果(N',M'_0)是活的，并且$^{\cdot}t_I\subseteq\{p\mid p\in P'\wedge(M'(p)>0)\}$，那么$(N,M_0)$与$(\overline{N}_T,\overline{M}_{T0})$都是活的。

证明：采用反证法。假设$\forall M_0''\in R(M'_0)$从Σ'中得到的Σ不活，即$\exists M\in R(M_0)$，$\exists t\in T$，$\forall\overline{M}\in R(M)$都有$\neg(\overline{M}[t>)$。由于$M_0''$在$\Sigma$上的投影为$M_0$，记$M_0[\sigma>M[\bar{\sigma}>\overline{M},\sigma,\bar{\sigma}\in T''$。现以$t_I\sigma_Tt_O$分别替换$\sigma,\bar{\sigma}$中的$\tilde{t}$，得$\sigma',\sigma''\in T''$。根据定义 4.7 和$\Sigma'$的活性知，$M'_0[\sigma'>M'[\bar{\sigma}'>\overline{M}'$，并且$M'$在$\Sigma$上的投影为$M$，$\overline{M}'$在$\Sigma$上的投影为$\overline{M}$，这样对应于$M$，$\exists M'\in R(M_0'')$，$\exists t'\in T''$(当$t\in T-\{\tilde{t}\}$时，$t'=t$，当$t=t'$时，$t'=t_I$)，使得对应于$\forall\overline{M}\in R(M)$有$\forall\overline{M}'\in R(M')$。由$\neg(\overline{M}[t>)$可推知，$\neg(\overline{M}'[t'>)$，从而$\Sigma'$不活，矛盾。因此$\exists M_0'',M_0'''\in R(M'_0)$，使得从$(N',M_0'')$中得到的$(N,M_0)$和从$(N',M_0''')$中得到的$(\overline{N},\overline{M}_{T0})$都是活的。又因为$^{\cdot}t_I\subseteq\{p\mid p\in P'\wedge(M'(p)>0)\}$，则从$(N',M'_0)$中得到的$(N,M_0)$和$(\overline{N}_T,\overline{M}_{T0})$都是活的。

定理 4.3 设(N,M_0)是由(N',M'_0)经 T-型子网化简操作得到的 Petri 网系统，如果(N,M_0)与$(\overline{N}_T,\overline{M}_{T0})$都是活的，那么$(N',M'_0)$是活的。

证明：$\forall t'\in T'$，则$t'\in T-\{\tilde{t}\}$或$t'\in T_T-\{t_I,t_O\}$或$t'\in\{t_I,t_O\}$。对$\forall M'\in R(M'_0)$，令$M'=[M,M_T]$，根据假定 4.1 有$M\in R(M_0)$和$M_T\in R(M_{T0})$。

(1)若$t'\in T-\{\tilde{t}\}$，由Σ的活性知，$\exists\overline{M}\in R(M)$，使得$\overline{M}[t'>$。记$M_0[\sigma_0>M[\sigma>\overline{M}[t'>$，其中，$\sigma_0,\sigma$为$\Sigma$的可引发变迁序列。

(1.1)若$\tilde{t}$不属于σ_0或σ的变迁集合，则根据定义 4.6 和定义 4.7，令$M'_0=[M_0,M_{T0}]$，$\overline{M}'=[\overline{M},M_{T0}]$，从而$\overline{M}\in R(M_0)$，$\overline{M}'\in R(M'_0)$，并且$M'_0[\sigma_0>M'[\sigma>\overline{M}'[t'>$，即$t'$在$\Sigma'=(N',M'_0)$中是活的；

(1.2)若$\tilde{t}$属于σ_0或σ中的变迁集合，不妨设$\tilde{t}$属于σ的变迁集合，且记为$M_0[\sigma_0>M[\sigma_1\ \tilde{t}\,\sigma_2>\overline{M}'[t'>$。由定义 4.6 及$\Sigma_T$的活性知，$M'_0[\sigma_0>M'[\sigma_1t_I\sigma_Tt_O\sigma_2>\overline{M}'[t'>$，其中$\sigma_T$是$T_T$上的步串，因此$t'$在$\Sigma'$中仍是活的。

(2)若$t'\in T_T-\{t_I,t_O\}$，由$M_0[\sigma>M$，即$[M_0,M_{T0}][\sigma'>[M_2,M_T]$，记$M_{T0}[\sigma_{T0}>M_T$。

(2.1)若t'不属于σ_{T0}的变迁集合，则由Σ_T的活性知，$\exists\overline{M}_T\in R(M_T)$使得$M_{T0}[\sigma_{T0}>M_T[\sigma_T>\overline{M}_T[t'>$，考虑到$\Sigma$的活性，因此$[M_0,M_{T0}][\sigma_0>[M_2,M_T][\sigma_{T0}>[M_2,M_T'][\sigma_T>[M_2,\overline{M}_T][t'>$，令$\overline{M}'=[M_2,\overline{M}_T]$，则有$M'_0[\sigma_0\sigma_{T0}>M'[\sigma_T>\overline{M}'[t'>$，因此$t'$在$\Sigma'$中是活的。

(2.2)若t'属于σ_{T0}的变迁集合，则有$[M_0,M_{T0}][\sigma_0t_I>[M_1,M_T][\sigma_{T0}>[M_1,M_{T1}][\sigma_1>[M_2,M_{T1}]$，根据$\Sigma_T$的活性，$\exists\sigma''$使得$[M_2,M_{T1}][\sigma''t_O>[M_3,M_{T0}]$。又根据$\Sigma$的活性知，$\exists\sigma_2$使得$[M_3,M_{T0}][\sigma_2t_I>[M_1,M_T]$，再根据$\Sigma_T$的活性，$\exists\sigma_T$使得$[M_1,M_T][\sigma_T>[M_1,\overline{M}_{T1}][t'>$，(其中$\sigma_T$是$\sigma_{T0}$的前缀子串)。亦即，$[M_0,M_{T0}][\sigma_0t_I\sigma_{T0}\sigma_1>[M_2,M_{T1}][\sigma''t_O\sigma_2t_I\sigma_T>[M_1,\overline{M}_{T1}][t'>$，即$M'_0[\sigma_0t_I\sigma_{T0}\sigma_1>M'[\sigma''t_O\sigma_2t_I\sigma_T>\overline{M}'[t'>$，因此$t'$在$\Sigma'$中是活的。

(3)若$t'\in\{t_I,t_O\}$不妨设$t'=t_I$，对$\forall M'\in R(M'_0)$，其中$M'=[M,M_T]$，$M\in R(M_0)$，$M_T\in R(M_{T0})$。

(3.1)若$M_T=\theta_T$，则由Σ的活性知，$\exists\sigma$使得$M[\sigma>\overline{M}[\tilde{t}>$，而$^{\cdot}\tilde{t}={}^{\cdot}t'$，因此在$\Sigma'$中$\exists\overline{M}'=[\overline{M},\theta_T]$，使得$M'_0[\sigma_0>M'[\sigma>\overline{M}'[t'>$，即$t'$在$\Sigma'$上是活的。

(3.2)若$M_T\neq\theta_T$，则由Σ_T的活性知，$\exists\sigma_T$使得$M_T[\sigma_T>M_{T1}$。由Σ的活性知，$\exists\sigma_0$使得$M_{T1}[t_O\sigma_o>\overline{M}[\tilde{t}>$，而$^{\cdot}\tilde{t}={}^{\cdot}t'$，从而在$\Sigma'$中有$M'_0[\sigma'_0>M'[\sigma_Tt_O\sigma_o>\overline{M}'[t'>$，从而$t'$在$\Sigma'$上是活的。

由(1)，(2)，(3)可知，t'在Σ'上是活的；由t'的任意性知，Σ'是活的。

定理 4.4 设(N,M_0)是由(N',M'_0)经 T-型子网化简操作得到的 Petri 网系统，如果(N',M'_0)是可回复的，并且$^{\cdot}t_I\subseteq\{p\mid p\in P'\wedge(M'(p)>0)\}$，那么$(N,M_0)$与$(\overline{N}_T,\overline{M}_{T0})$都是可回复的。

证明：采用反证法，假设(N,M_0)不是可回复的，则$\exists M_1\in R(M_0)$，使得$M_0\notin R(M_1)$，根据定义 4.6 和定义 4.7，$\exists M_1'=[M,M_T]$，使得$M'_0\notin R(M_1')$。这与题设(N',M'_0)是可回复的相矛盾。因此，$\exists M_0'',M_0'''\in R(M'_0)$，使得从$(N',M_0'')$中得到的$(N,M_0)$和从$(N',M_0''')$中得到的$(\overline{N}_T,$

$\bar{M}_{T0}$)都是可回复的。又因为$^{\cdot}t_l\subseteq\{p\mid p\in P'\wedge(M'(p)>0)\}$，则从$(N',M'_0)$中经精细化操作得到的$(N,M_0)$和$(\bar{N}_T,\bar{M}_{T0})$都是可回复的。

定理 4.5 设(N,M_0)是由(N',M'_0)经 T-型子网化简操作得到的 Petri 网系统，如果(N,M_0)与$(\bar{N}_T,\bar{M}_{T0})$都是可回复的，那么(N',M'_0)是可回复的。

证明：$\forall M'\in R(M'_0)$，根据定义 4.6 和定义 4.7，$M'_0=[M_0,M_{T0}]$，$M'=[M,M_T]$。因为(N,M_0)是可回复的，则$\forall M\in R(M_0)$，$M_0\in R(M)$。又因为$(\bar{N}_T,\bar{M}_{T0})$是可回复的，则$\forall M_T\in R(M_{T0})$，$M_{T0}\in R(M_T)$。显然，$M'_0\in R(M')$，因此(N',M'_0)是可回复的。

5 应用

本节将应用前述 Petri 网化简操作方法对一个具体的柔性制造系统进行验证。

该柔性制造系统包括 3 个过程：一个工作站（用于装配）和两个制造中心（用于加工制造）。制造中心 1 与工作站 WS 共享机器人R_1，制造中心 2 与工作站 WS 共享机器人R_2。系统按如下方式运行：

(1)制造中心 1 用于将半成品零件在机器 M_1 上进行再加工。在运行过程中，零件被机器人 R_1 固定在货板上，再放置在机器 M_1 上。零件在 M_1 上加工，加工完成后，R_1 负责卸下成品零件，并返回货板。

(2)制造中心 2 用于将原零件（未经加工）先在机器 M_3 上加工，然后在机器 M_4 上进行再加工。零件被机器人 R_2 固定在货板上，再放置在机器 M_3 上。在 M_3 上加工完成后，机器人 R_2 从 M_3 上卸掉中间件并放入缓冲器 B 中。在机器 M_4 上加工过程中，中间件 R_2 被从缓冲器 B 中取出，固定在 M_4 上加工。加工完成后，R_2 负责卸下成品零件，并返回货板。

(3)工作站 WS 用于装配。在装配过程中，机器人 R_1、R_2 和机器 M_2 合作完成装配工作。

(4)在一般的共享机器人操作的系统中，不考虑改变机器人本身的配置。这里，我们考虑一个更一般的情况。当机器人完成一个工种，而要转去完成另一个工种时，需要经过一些中间处理（比如、清洁、润滑、更换相关配件等）。中间处理结束后，再去执行另一工种。这样做的好处是，一个机器人能完成尽可能多的相近工种。

(5)假定在输入处总有未加工的零件供应，并且制造出的成品总能被及时移走。

注：在用 Petri 网描述柔性制造系统时，每个过程被抽象为一个库所，每个变迁表示一个过程的开始或（和）结束。

图 3（引自文[15]，并做大量改动而得，它在文[15]中用于解释资源共享的概念，这里用于验证化简操作的结果）给出该系统的 Petri 网模型$\Sigma=(N,M_0)$。

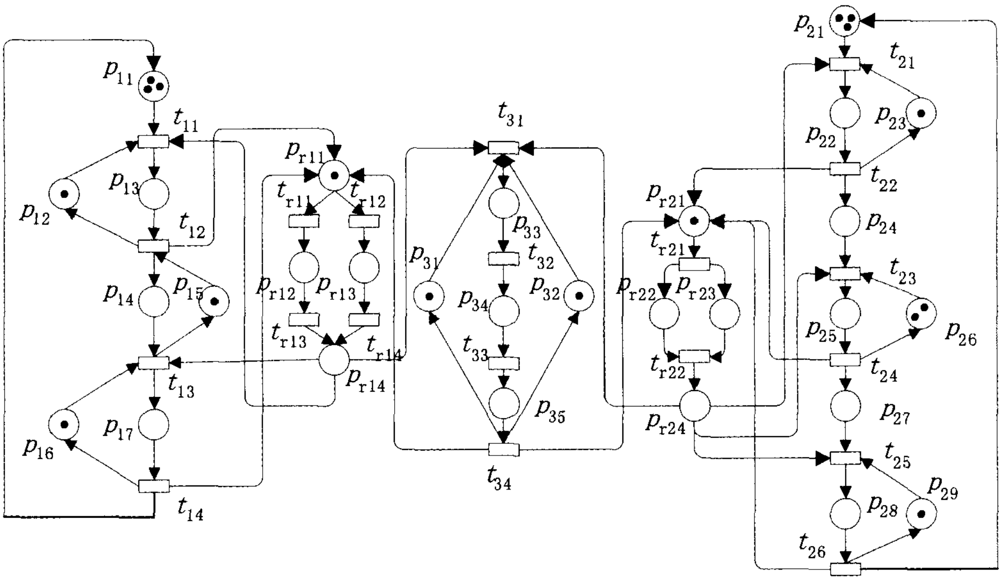

图 3 原系统 Petri 网模型

其中库所和变迁的含义为：

p_{11}：货板可用，半成品零件可用；

t_{11}：开始执行 p_{13}；

p_{12}：申请使用机器人 R_1；

t_{12}：完成 p_{13}，开始执行 p_{14}；

p_{13}：获得 R_1 的使用权，且 R_1 将货板固定在机器 M_1 上；

t_{13}：完成 p_{14}，开始执行 p_{17}；

p_{14}：零件在机器 M_1 上再加工；

t_{14}：完成 p_{17}；

p_{15}：机器 M_1 可用；

p_{16}：申请使用机器人 R_1；

p_{17}：卸下成品，返回货板；

p_{21}：货板可用，原零件（未经加工）可用；

t_{21}：开始执行 p_{22}；

p_{22}：机器 M_3 负载，加工货板上的零件；

t_{22}：完成 p_{22}，开始执行 p_{24}；

p_{23}：机器 M_3 可用；

t_{23}：完成 p_{24}，开始执行 p_{25}；

p_{24}：准备放入缓冲器 B 的中间件；

t_{24}：完成 p_{25}，开始执行 p_{27}；

p_{25}：机器人 R_2 将中间件放入缓冲器 B；

t_{25}：完成 p_{27}，开始执行 p_{28}；

p_{26}：缓冲器 B 可用；

t_{26}：完成 p_{28}；

p_{27}：中间件可用；

p_{28}：在 M_4 上加工，然后机器人 R_2 从机器 M_4 上卸下成品，并返回货板；

p_{29}：机器 M_4 可用；

p_{31}:机器 M_2 可用;

t_{31}:开始获得机器人 R_1 和 R_2;

p_{32}:申请使用机器人 R_1 和 R_2;

t_{32}:在工作站 WS 中开始第一步装配;

p_{33}:WS 获得机器人 R_1 和 R_2 的使用权;

t_{33}:在工作站 WS 中开始最后一步装配;

p_{34}:在 WS 中进行第一步装配;

t_{34}:完成在工作站 WS 中的装配;

p_{35}:在 WS 中进行最后一步装配;

p_{r11}:机器人 R_1 可用;

t_{r11}:开始清洁机器人 R_1 并润滑;

p_{r12}:对机器人 R_1 进行清洁并加润滑油;

t_{r12}:(为适应不同工种)开始为 R_1 更换有关配件;

p_{r13}:为 R_1 更换有关配件;

t_{r13}:清洁、润滑 R_1 结束;

p_{r14}:机器人 R_1 备用;

t_{r14}:为 R_1 更换配件结束;

p_{r21}:机器人 R_2 可用;

t_{r21}:开始为 R_2 清洁、润滑和更换配件;

p_{r22}:对机器人 R_2 进行清洁并加润滑油;

t_{r22}:为 R_2 清洁、润滑和更换配件结束;

p_{r23}:R_2 为更换有关配件;

p_{r24}:机器人 R_2 备用。

对网系统(N,M_0)(图 3)的验证分以下 3 步:

第一步,将(N,M_0)中的 P-型子网系统(N_{P1},M_{P10})(由$\{p_{r11},p_{r12},p_{r13},p_{r14},t_{r11},t_{r12},t_{r13},t_{r14}\}$生成)、$(N_{P2},M_{P20})$(由$\{p_{r21},p_{r22},p_{r23},p_{r24},t_{r21},t_{r22}\}$生成)分别化简为带有标识的库所 p_{r1},p_{r2},得到网系统(N_1,M_1)(图 4)。易见,对(N,M_0)的 P-型子网化简满足定理 3.1~3.5 的条件。又因为两个 P-型闭网系统(图示未给出)都是有界的、活的和可回复的,所以根据定理 3.1~3.5 知,(N,M_0)是有界的、活的和可回复的,当且仅当(N_1,M_1)是有界的、活的和可回复的。

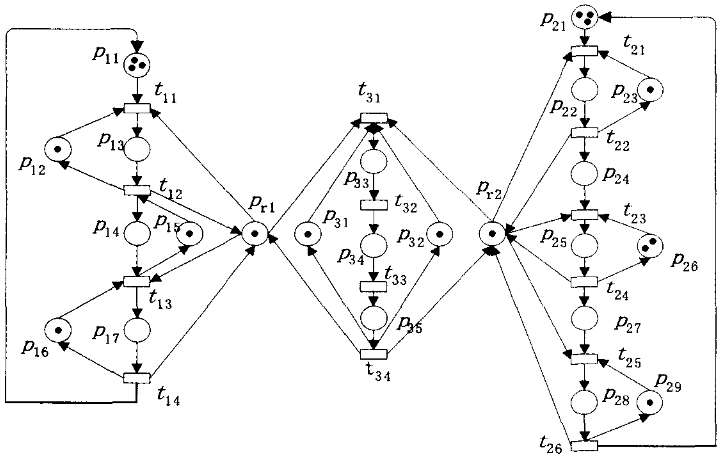

图 4 将 P-型子网化简后的网系统

第二步,将(N_1,M_1)(图 4)中的 T-型子网系统(N_{T1},M_{T10})(由$\{p_{12},p_{13},t_{11},t_{12}\}$生成)、$(N_{T2},M_{T20})$(由$\{p_{16},p_{17},t_{13},t_{14}\}$生成)、$(N_{T3},M_{T30})$(由$\{p_{31},p_{32},p_{33},p_{34},p_{35},t_{31},t_{32},t_{33},t_{34}\}$生成)、$(N_{T4},M_{T40})$(由$\{p_{22},p_{23},t_{21},t_{22}\}$生成)、$(N_{T5},M_{T50})$(由$\{p_{25},p_{26},t_{23},t_{24}\}$生成)、$(N_{T6},M_{T60})$(由$\{p_{28},p_{29},t_{25},t_{26}\}$生成)分别化简为变迁 t_1',t_2',t_3',t_4',t_5',t_6',得到网系统(N_2,M_2)(图 5)。对(N,M_0)的 T-型子网化简,满足定理 4.1~4.5 的条件。又因为 6 个 T-型闭网系统(图示未给出)都是有界的、活的和可回复的,所以根据定理 4.1~4.5 知,(N_1,M_1)是有界的、活的和可回复的,当且仅当(N_2,M_2)是有界的、活的和可回复的。

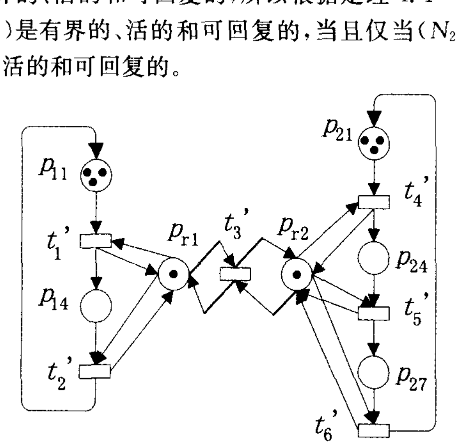

图 5 再将 T-型子网化简后的网系统

第三步,删除(N_2,M_2)中满足${}^\cdot p=p^\cdot$且$M_2(p)>0$的库所(即 p_{r1},p_{r2}),得到网系统(N_3,M_3)(图 5)。因为含有托肯的库所 p_{r1},p_{r2}在(N_2,M_2)中包含自环,删除这些库所及与它们相连的弧,并不影响发射序列和托肯分布,所以(N_2,M_2)是有界的、活的和可回复的,当且仅当(N_3,M_3)是有界的、活的和可回复的。

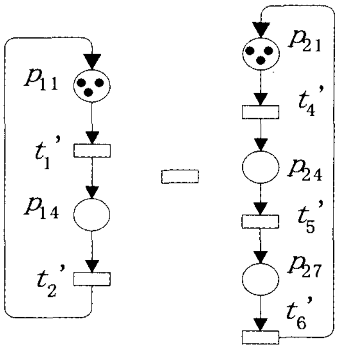

图 6 去掉自环后的网系统

由以上 3 步可得,复杂的柔性制造系统模型(N,M_0)(图 3)是有界的、活的和可回复的,当且仅当(N_3,M_3)是有界的、活的和可回复的。因为(N_3,M_3)是两个含初始标识的环路和一个独立的变迁,它显然是有界的、活的和可回复的[12],所以

(下转第 286 页)

实时比对，并分别设置其容差范围，超出相应范围即可报警，并对监控画面进行硬盘录像。在台标轮廓比较算法中不存在延迟处理，视频捕获卡每秒捕获25帧图像，我们每秒钟只取其中的两帧来比较就可以了。

系统中还显示抓取的台标轮廓和实时画面中台标轮廓中同一行的YUV分量波形，通过观察两种波形是否重叠来调节实时画面的亮度和色度，使其和抓取的台标轮廓图像的亮度和色度一致，保证台标轮廓比较的准确性。

音频监控主要是对同一时刻上下行音频的分贝进行比较，如果存在延迟，则加上延迟。在比较过程中，如果两路对比信号中某一路存在背景噪声或噪声不一致，这可能引起音频信号对比结果出错，我们可以对其进行静噪。保证对比结果的准确性。同时可以对上下行音频信号的左右声道分别进行负载调节，从而保证上下行音频对比时音量的一致性，使比较结果更加准确。

当比较结果超出容差范围的时候自动进行声光报警，屏蔽非法画面，切换通道。同时进行硬盘录像，可以回放检索，以便事后进行分析。

另外，透明台标轮廓的比较主要是进行轮廓相似度的计算，其算法和不透明台标有很大的相似性，这里就不具体介绍了。

2.3 资源管理

为了解决CPU利用率的问题，提高系统的运行效率，我们采用了多线程编程技术。多线程机制是Windows95推出的，CPU通过分时在各个线程间频繁切换，使系统看起来好像是有多道程序同时运行。

系统首先采用一个主控线程控制用户界面，以便用户随时控制系统的运行，或者随时修改系统中有关参数的设置，如画面亮度调节、音量的控制、容差范围设置等。对于实时视频采集显示、实时台标轮廓比较算法、YUV分量波形实时显示、硬盘录像等功能我们分别建立其线程来实现。系统中各线程采用不同优先级，优先级高的线程优先获得CPU时间片，对视音频的实时采集、量化、处理和台标轮廓比较具有最高优先级，而其他线程具有同等优先级，保障了监控系统的实时性和准确性。

另外，服务器端到客户端的数据传输是实时的，并且视音频需要同步，在处理系统数据时先对其进行缓冲后再进行其它处理，一定程度上解放了CPU资源，提高了CPU使用效率。

多线程编程实现简单，系统开销小，大大增强了系统的可用性和稳固性。实验表明，采用多线程技术使系统具有更强的数据处理能力，高效地实现了系统资源的管理。在监控点正常运行的情况下，监控点CPU占用7%左右，如果系统出现报警并进行硬盘录像等操作的时候，CPU资源占用率将达到50%，这是系统需要CPU资源最多的情况，因此采用多线程编程方式可以提高整个软件的工作效率和资源利用率。

总结 本系统对各不透明台标电视台分别进行了测试，在电视画面出现质量改变或者出现亮度、色度差异时，监控系统能够实时、准确地报警，实现了地球站对卫星节目电视播出质量、非法信号攻击的监控。并且我们在晚上7点播放新闻联播的时候，出现的台标重叠情况也进行了测试，各电视台台标都是重叠在中央一台的上面，对不透明台标监控系统不造成什么影响，系统能够稳定地运行。该系统成功地实现了对地球站电视节目的监控和报警，并且在检测到电视节目出现质量问题或者受到非法信号攻击的时候能够采取有效措施进行事后处理。研究和开发其体系结构和其中采用的各项技术具有重要的意义和前景。

参考文献

1 肖建荣，胡建凌，徐盛．一种远程视频监控系统的研究和实现[J]．电视技术，2004(6)：74～76
2 朱之芹，杨林安，郭宇．多线程在数字图像实时接收中的应用[J]．电视技术，2001(7)：20～21
3 张重阳，娄震，杨静宇．基于轮廓和统计特征的手写体数字识别[J]．计算机工程与应用，2004(9)：83～84
4 张理慧，张宇．基于DirectShow的视频捕获[J]．计算机工程，2004(10)：131～133
5 武雅丽．利用Visual C++6.0实现多线程编程[J]．现代电子技术，2003(8)：45～47
6 杜恩祥，左宪章．基于多线程机制应用程序中线程间通信和同步问题的解决[J]．无线电通信技术，2000(4)：51～52

（上接第240页）

(N,M_0)是有界的、活的和可回复的。

结论和下一步的研究工作 本文讨论了Petri网的化简操作的动态性质保持问题及其在系统验证中的应用。主要贡献在于：针对柔性制造系统的验证问题提出了两种化简操作；给出了化简操作后得到的Petri网保持活性、有界性和可回复性的充分条件或充要条件。文中的应用实例进一步展示了该方法的实际价值。本文的结果可为复杂大系统的设计和验证提供有力保证。

下一步的研究工作是进一步推广化简操作满足动态性质的保持性条件，并研究化简操作对其它性质（如公平性等）的保持性问题。

参考文献

1 Desel J. Reduction and design of well-behaved concurrent systems. Lecture Notes on Comput Sci.，1990，458：166～181
2 Esparza J. Reduction and synthesis of live and bounded free choice Petri nets. Inform Comput，1994，114(1)：50～87
3 Desel J，Esparza T. Free choice Petri nets. Cambridge：Cambridge University Press，1995
4 Esparaza J，Silva M. On the analysis and synthesis of free choice systems. Lecture Notes in Comput Sci.，1990，483：186～243
5 Esparza J，Silva M. Compositional synthesis of live and bounded free choice Petri nets. Lecture Notes in Comput Sci.，1991，527：172～187
6 Jiao L，Cheung T Y，Lu W M. On liveness and boundedness of asymmetric choice nets. Theret Comput Sci.，2004，311：165～197
7 Nakada K，Yoneyama T. A method to abstract a stochastic Petri net. Mathematical and Computer Modeling，2000，31：251～260
8 Ceška M，Hasa L，Vojnar T. Partial-order reduction in model checking object-oriented Petri nets. Lecture Notes in Computer Science，Springer-verlag Gmbh，2004，2089：266～278
9 Sloan R H，Buy U. Reduction rules for time Petri nets. Acta Informatica 1996，33：687～706
10 Garavel H，Serwe W. State space reduction for process algebra specifications. Lecture Notes in Computer Science，Springer～Verlag GmbH，2004，3116：164～180
11 Bingham J，Hu A J. Empirically efficient verification for a class of infinite-state systems. Lecture Notes in Computer Science，Springer-Verlag GmbH，2005，3440：77～92
12 Murata T. Petri nets：properties，analysis，and applications. Proc. IEEE，1989，77(4)：541～580
13 Peterson J L. Petri net theorey and the modeling of systems. Englewood Cliffs，New York：Prentice-Hall Inc，1981
14 Resig W. Petri nets. EATCE Monographs on Theoretical Computer Science. New York：Spring-Verlag，1985，4
15 Zhou M，DiCesare F. Petri net synthesis for discrete event control of manufacturing systems. Dordrecht：Kluwer Academic publishers，1993