マルチメディア通信と分散処理 29－3
（１９８６．５．１５）

ペトリネットモデルによる分散システムのモデル化と動作検証

山田智彦† 辻孝吉# 熊谷貞俊# 児玉慎三#

†ダイキン工業株式会社 #大阪大学工学部電子工学科

分散システムの動作解析のためのモデルとしてペトリネットがある。ペトリネットは、ネットの一部をその部分をより詳しく表現したネットで逐次置き換えることによりトップダウン形式の階層表現が容易である。このとき、詳細化前のネットのライブ性や有界性等の性質が、詳細化後のネットでも保存されるかということが問題となる。

本文では、この問題を保証する元のネット及び代置するネットの構造を明らかにし、また、本文で与える階層表現方法が大規模分散システムの動作検証に有効であることを示す。

" Modelling and verification of distributed systems by petri nets"

Tomohiko YAMADA†, Kohkichi TSUJI#, Sadatoshi KUMAGAI#,
Shinzo KODAMA#

†Daikin Industries Ltd.
#Electronic Engineering of Osaka University

Distributed systems such as computer systems, production systems and socio-economic systems can be modelled by petri nets. Petri net model can express cocurrency and synchronization which are the main characteristic features of distributed systems. Petri nets are also suited to a hierarchical representation of a large scale distributed systems. In this paper,a method of stepwise refinement of a set of transitions are proposed for the application to the top-down design. A well-behavedness of a subnet is defined so that the augmented model by this net is guaranteed to preserve fundamental properties, such that liveness, boundedness,and safeness.

１．まえがき

通信プロトコル、ＦＭＳ、ＯＡシステム等のように、複数の離散的事象が非同期・並列に進行する大規模な分散システムに対して、その体系的な設計手法の確立が望まれている。

大規模システムの設計においては、いくつかのサブシステムを組み合わせて一つの統合されたシステムを構成する方法や、システムの大枠から設計をおこない各部を順次具体化してゆく方法がとられる。このような手法を用いた柔軟かつ迅速な設計の自動化を図るためには、各設計段階における仕様記述・解析手法の形式化・体系化が重要である。

ペトリネット[1]は分散システムの動作解析のためのモデルとして有効性が広く認識されつつあるが、これをシステムの仕様記述から解析・検証に至るまで一貫して用いることにより、上記の目的を達成する設計手法が構築し得るものと期待される。

ペトリネットは、プレース、トランジション（図中、それぞれ丸印○、四角□で表される）と呼ばれる２種類の節点集合とそれらを結合する有向枝で構成される２部グラフで表される（図１）。通常、プレースは事象生起に関するシステムの条件を表し、トランジションは離散事象の生起と完了を表現する。この２部グラフはシステムの静的構造を表し、また、システムの動的挙動はトランジションを発火させプレース中に置かれるトークン（図中、黒丸●）を移動することによって表現される。

ペトリネットモデルの利点の一つは、トランジションをそのトランジションが表す動作を詳細表現した他のペトリネットで逐次置き換えることにより、トップダウン形式のシステム表現が可能となることである。

このような階層表現において問題となるのは、詳細化する前のペトリネットの動作が詳細化後のペトリネットにおいても完全に保存されているかどうかということである。逆に言うと、このような性質を保証する代置ネットの構造と初期トークンの配置を明らかにしておく必要がある。単一トランジションの詳細化に関して上記の問題はValette[2]及び鈴木、村田[3]らによって考察され、逐次詳細化されるシステムのライブ性、有界性などの性質が、元のシステムの性質と代置ネットの

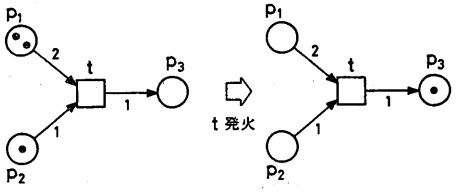


図１　ペトリネット

性質を調べることにより検証し得ることが判っている。

いま、単一トランジションを詳細化することによってシステムを階層表現しようとする。このとき、複数のトランジションを詳細化した各々のサブネットの動作に相互関係が生ずる場合、それらのサブネットを接続する必要がある。換言すれば、階層化のレベルに従いシステム各部の相互干渉が陽に現れてくるのである。従って、上位レベルで互いに関連のありそうな複数のトランジションをまとめて一つのネットで詳細化できれば、より効率的な階層表現が可能となる。

本文では、階層表現における層間での性質の保存性に関する[2]や[3]の結果を、システムの任意の部分（トランジションの集合）の詳細化に対して一般化する。また、この階層表現によって大規模分散システムの動作検証が容易になることを示す。

２．ペトリネット

まず本稿で用いる諸記号及びペトリネットの定義をする。

Z^+：非負整数

S^*：集合Ｓの要素から構成されるすべての系列の集合

S^+：$S^* \setminus \{\lambda\}$

任意に大きな数ω

ω：次の式を満足する数

$$\forall n \in Z\ ;\ \omega > n,\ \omega \pm n = \omega,\ \omega \geq \omega$$

$\sigma \in S^*$，$s \in S$とする。

$|\sigma|$　：系列σの長さ

$\#(\sigma, s)$：系列σ上でｓが現れる回数

$\sigma \in S^*$，$S' \subseteq S$とする。

$\delta(\sigma, S')$：系列σからＳ'の要素のみを取り出して構成される部分系列

定義１（ペトリネット）

ペトリネットは次の５項組で構成される。

$N = (P, T, IN, OUT, M_0)$

P　：プレースの集合
T　：トランジションの集合
（$P \cap T = \phi$）
IN　：$T \times P \rightarrow Z^+$，トランジションの入力枝重み関数
OUT：$T \times P \rightarrow Z^+$，トランジションの出力枝重み関数
M_0　：$P \rightarrow Z^+ \cup \{\omega\}$，初期マーキング

定義２（発火条件）
ペトリネットN＝(P,T,IN,OUT,M_0)において、
$\forall p \in P$；$IN(t,p) \leqq M(p)$
であるとき、トランジションtはマーキングMで発火可能であるという。

定義３（マーキング遷移則）
マーキングM_1は、マーキングM_1で発火可能なトランジションtが発火すると、
$\forall p \in P$；$M_2(p) = M_1(p) - IN(t,p) + OUT(t,p)$
で定まるマーキングM_2に遷移する。このとき、$M_1[t>M_2$と表す。

定義４（可到達性）
マーキング$M_1, M_2, \cdots, M_{n+1}$に対し、$M_i[t_i>M_{i+1}$ $(i=1,2,\cdots,n)$となるトランジション$t_1, t_2, \cdots, t_n$が存在するとき、
$\sigma = t_1 t_2 \cdots t_n$
を発火系列という。マーキングM_{n+1}はマーキングM_1から可到達であるといい、$M_1[\sigma>M_{n+1}$で表す。

また、集合R(N)，L(N)を次のように定める。
R(N)：ペトリネットNにおいて、初期マーキングM_0から可到達なマーキングの集合
L(N)：ペトリネットNにおいて、初期マーキングM_0から始まるすべての発火系列の集合

ペトリネットはその構造及び初期マーキングの配置によって、有界性、安全性、ライブ性等の性質を持つ。これらは、システムを解析する上で重要な性質である。

定義５（有界性）
ペトリネットN＝(P,T,IN,OUT,M_0)は、
$\forall M \in R(N), \forall p \in P$；$\exists k \in Z^+$，$M(p) \leqq k$
となるとき、有界である。

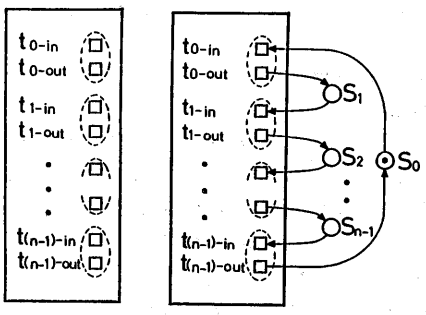
定義６（安全性）
ペトリネットN＝(P,T,IN,OUT,M_0)は、
$\forall M \in R(N)$，$\forall p \in P$；$M(p) \leqq 1$
となるとき、安全である。

定義７（ライブ性）
(1)ペトリネットN＝(P,T,IN,OUT,M_0)のなかのトランジションt∈Tは、
$\forall \sigma \in L(N)$；$\exists \sigma' \in T^*$，$\sigma\sigma' t \in L(N)$
となるとき、ライブである。
(2)すべてのトランジションt∈Tがライブであるとき、ペトリネットNはライブであるという。

３．well-behavedness

ペトリネットN＝(P,T,IN,OUT,M_0)に対し、入力トランジションの集合及び出力トランジションの集合
$T_{in} = \{t_{i\text{-}in} \mid 0 \leqq i \leqq n-1,\ i \in Z^+\}$ $(T_{in} \subset T)$
$T_{out} = \{t_{i\text{-}out} \mid 0 \leqq i \leqq n-1,\ i \in Z^+\}$,
$(T_{out} \subset T)$, $(T_{in} \cap T_{out} = \phi)$
を与える(図２(a))。これらの入出力トランジションの間に、次に示す規則に従いある初期トークンを持ったテストプレースの集合
$S = \{s_i \mid 0 \leqq i \leqq n-1,\ i \in Z^+\}$
を接続して得られるペトリネットをN_B＝(P_B, T_B, IN_B, OUT_B, M_{B0})とする(図２(b))。

[N_Bの構成規則]
$N_B = (P_B, T_B, IN_B, OUT_B, M_{B0})$
$P_B = P \cup S$,
$T_B = T$,
$IN_B(t,p)$
$= 1$　；$t=t_{i\text{-}in}$ and $p=s_i$
$= 0$　；$t=t_{i\text{-}in}$ and $p=s_j$ $(i \neq j)$
$= IN(t,p)$；$p \in P$,



(a)入力トランジションを与えたペトリネット　　(b)テストプレースの接続
図２

$OUT_B(t,p)$

$= 1$; $t=t_{i\text{-}out}$ and $p=s_{i+1}$ $(s_n=s_0)$

$= 0$; $t=t_{i\text{-}out}$ and $p=s_j$ $(j\neq i+1)$

$= OUT(t,p)$; $p\in P$,

$M_{B0}(p) = 1$; $p=s_0$

$= 0$; $p=s_i$ $(i\neq 0)$

$= M_0(p)$; $p\in P$

定義８（well-behavedness）

ペトリネット$N=(P,T,IN,OUT,M_0)$は、次の条件(1)，(2)を満たすときトランジション$T_{in}\subset T$, $T_{out}\subset T$ $(T_{in}\cap T_{out}=\phi)$に関してwell-behavedであるという。

(1) $N_B=(P_B,T_B,IN_B,OUT_B,M_{B0})$において$t_{0\text{-}in}$はライブである。

(2) $\forall\sigma\in L(N_B)$;

(a) $\forall i,j,i<j$;

$\#(\sigma,t_{i\text{-}in}) - \#(\sigma,t_{j\text{-}in}) = 1$ or 0

(b) $\forall i,j,i<j$;

$\#(\sigma,t_{i\text{-}out}) - \#(\sigma,t_{j\text{-}out}) = 1$ or 0

(c) $\forall i$;

$\#(\sigma,t_{i\text{-}in}) - \#(\sigma,t_{i\text{-}out}) = 1$ or 0

well-behavednessはトランジションの発火順序に関する性質である。

性質１

ペトリネット$N=(P,T,IN,OUT,M_0)$がトランジション$T_{in}\subset T$, $T_{out}\subset T$ $(T_{in}\cap T_{out}=\phi)$に関してwell-behaved条件(2)を満たすことは、次のことと等価である。

すべての発火系列$\sigma\in L(N_B)$に対し、$T_{in}\cup T_{out}$の要素から構成される部分系列を

$\delta(\sigma,T_{in}\cup T_{out})=a_0a_1a_2a_3\cdots a_{2k}a_{2k+1}$

とするとき

$a_{2k}=t_{k(mod\ n)\text{-}in}$, $a_{2k+1}=t_{k(mod\ n)\text{-}out}$

である。

（証明）省略。

性質２

ペトリネット$N=(P,T,IN,OUT,M_0)$が、トランジション$T_{in}\subset T$, $T_{out}\subset T$ $(T_{in}\cap T_{out}=\phi)$に関してwell-behavedであるならば、

$\delta(\sigma,T_{in}\cup T_{out})$が無限系列となる発火系列$\sigma\in L(N_B)$が存在する。

（証明）ライブの定義より明らかである。

４．トランジションの詳細化

トランジションを他のペトリネットで置き換えるときの規則を与える。

ペトリネット$N=(P,T,IN,OUT,M_0)$の中のトランジションの集合

$T_s=\{t_i \mid 0\leqq i\leqq n-1,\ i\in Z^+\}$ $(T_s\subset T)$

を、入出力トランジション

$T_{in}=\{t_{i\text{-}in} \mid 0\leqq i\leqq n-1,\ i\in Z^+\}$ $(T_{in}\subset T')$,

$T_{out}=\{t_{i\text{-}out} \mid 0\leqq i\leqq n-1,\ i\in Z^+\}$ $(T_{out}\subset T')$, $(T_{in}\cap T_{out}=\phi)$

が与えられたペトリネット$N'=(P',T',IN',OUT',M_0')$で置き換えて得られるペトリネットを$N''=(P'',T'',IN'',OUT'',M_0'')$とする。

（図３）

$P''= P\cup P'$,

$T''= (T\cup T')\setminus T_s$,

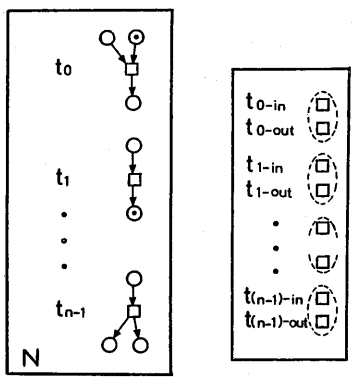
$IN''(t,p)$

$= IN(t,p)$; $t\in T\setminus T_s$ and $p\in P$,

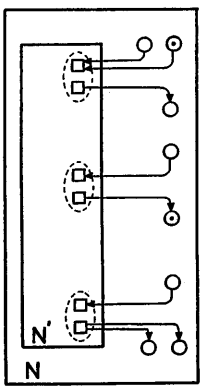
$= IN'(t,p)$; $t\in T'$ and $p\in P'$,

$= IN(t_i,p)$; $t=t_{i\text{-}in}$ and $p\in P$,

$= 0$; otherwise,



(a)詳細化される トランジション集合　(b)代置ネット N'



N''

図３　(c)詳細化ネット

$OUT''(t,p)$

$= OUT(t,p)$; $t \in T \setminus Ts$ and $p \in P$,

$= OUT'(t,p)$; $t \in T'$ and $p \in P'$,

$= OUT(t_i,p)$; $t = t_{i\text{-}out}$ and $p \in P$,

$= 0$; otherwise,

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

$= M_0'(p)$; $p \in P'$

以上の詳細化の規則に従ってシステムがどのように階層表現されるかを、ロボットアームの動作をモデル化することによって示す。

図４は、同じ仕様を持つ２基のロボットアームが部品をＰ点からＳ点へ運ぶシステムである[4]。

［動作仕様］

・Ｐ点に部品が置かれるとアームが作動し部品をＱ点へ運ぶ。

・Ｑ点に部品が置かれるとアームが作動し部品をＳ点へ運ぶ。

これをモデル化すると図５(a)になる。

次にトランジションA1、A2が表すアーム単体の動作を具体的にモデル化する(図５(b))。

［アーム１基の動作仕様(１サイクル)］

・初期状態　位置：ｘ，ハンド：開

・ｗ点に部品が置かれると下降(ｘ→ｗ)する。

・ハンドを閉じ部品をつかむ。

・ｗ→ｘ→ｙ→ｚの経路に従って移動する。

・ハンドを開き部品をｚ点に置く。

・ｚ→ｙ→ｘの経路に従って初期位置に戻る。

トランジションA1in，A1outが、A1を詳細化するときの入力トランジション、出力トランジションとなる。A2についても同様である。詳細化したネット(図５(c))はアームの動作仕様を満足している。

トランジションＵ，Ｄが表すアームの昇降は１個のモータで駆動される。モータのモデルは図６(a),(b)が考えられる。共に単体では正常に動作するモータである。Ｕin,Ｄinを Ｕ,Ｄの入力トランジション、Ｕout,Ｄoutを出力トランジションとして置き換えると図

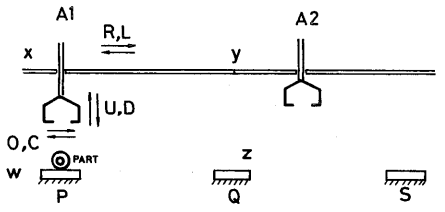
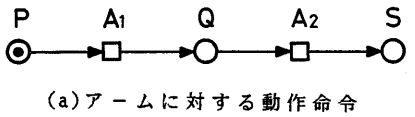
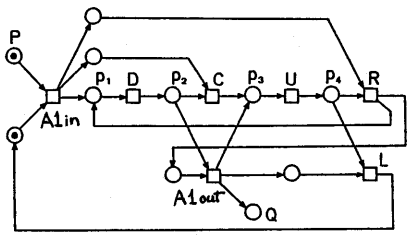


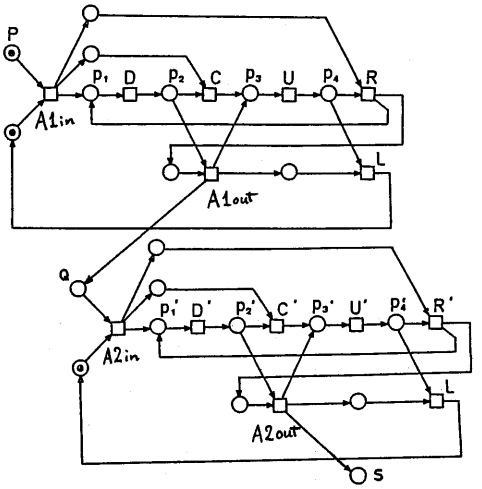
図４　ロボットアーム運搬システム



(a)アームに対する動作命令

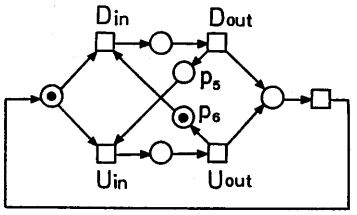


(b)アーム１基の動作

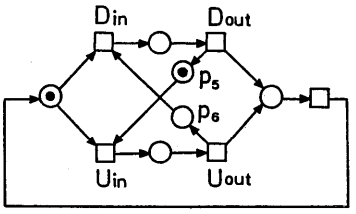


(c)アーム２基の動作

図５　ロボットアームのモデル化



(a)



(b)

図６　モータのモデル

7(a),(b)になる。図7(a)はアーム1基の仕様を満足しているが、図7(b)はプレースp6にトークンが置かれないためDinが発火できずデッドロックとなってしまう。即ち、元のネットのライブ性が詳細化後のネットでは保存されなくなる。従って、図7(b)のネットは代置するネットとして用いることが出来ないことになる。

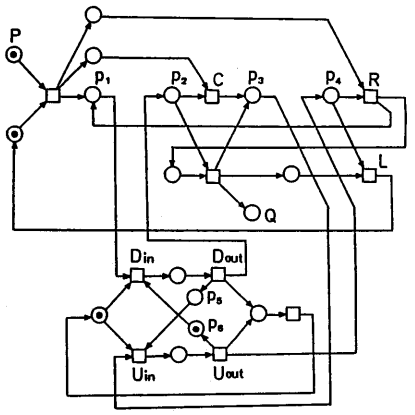
詳細化に伴うこの例におけるような問題を解決するために、本文では代置するネットはwell-behavedであるとし、また、詳細化を行うトランジションの集合に以下のような発火順序に関する条件を設ける。

[変換条件]

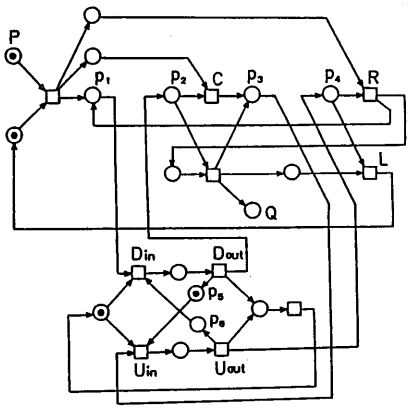
ペトリネットN=(P,T,IN,OUT,M_0)の中の詳細化されるトランジションの集合Ts⊆Tに対して、変換条件を次のように定める。

$\forall \sigma \in L(N), \forall i,j,i<j$;

$\#(\sigma,t_i) - \#(\sigma,t_j) = 1$ or 0



(a)正常動作



(b)デッドロック

図7　詳細化したアームのモデル

性質3

ペトリネットN=(P,T,IN,OUT,M_0)の中のトランジションの集合$Ts=\{t_i \mid 0 \leq i \leq n-1, i \in Z^+\}$,(Ts⊆T)が変換条件を満たすことは、すべての発火系列σ∈L(N)に対しTsの要素から構成される部分系列を$\delta(\sigma,Ts)=a_0a_1\cdots a_k$とするとき$a_k=t_{k(\mathrm{mod}\ n)}$となることと等価である。

(証明)省略。

5．性質の保存

前節で示した規則に従って、ペトリネットNのトランジション集合TsをサブネットN'で詳細化し新たにペトリネットN"を構成したとする。このときNとN"の間の性質の保存性に関する基本的な結果を以下に示す。

まず、N"のトランジションT"に関する発火系列をNのトランジションTに関する発火系列に変換する関数φ、およびN'のトランジションT'に関する発火系列に変換する関数ψを定義する。

定義10

(1) $\phi : (T")^* \rightarrow T^*$

(a) $\phi(\lambda) = \lambda$

(b) σ"∈(T")*について

$\phi(\sigma"t) = \phi(\sigma")$; $t \in T' \setminus T_{in}$

$= \phi(\sigma")t_i$; $t = t_{i-in} \in T_{in}$

$= \phi(\sigma")t$; $t = T \setminus Ts$

(2) $\psi : (T")^* \rightarrow (T')^*$

(a) $\psi(\lambda) = \lambda$

(b) σ"∈(T")*について

$\psi(\sigma"t) = \psi(\sigma")$; $t \in T \setminus Ts$

$= \psi(\sigma")t$; $t \in T'$

補題1

すべてのσ"∈L(N")について、

$\delta(\sigma", T_{in} \cup T_{out}) = a_0a_1a_2a_3\cdots a_{2k}a_{2k+1}$

ただし、$a_{2k}=t_{k(\mathrm{mod}\ n)-in}$,

$a_{2k+1}=t_{k(\mathrm{mod}\ n)-out}$

である。

(証明)

σ"=λのとき$\delta(\sigma", T_{in} \cup T_{out})=\lambda$

$|\sigma_1"|=d$となるすべてのσ1"∈L(N")について成立すると仮定する。

σ1"t∈L(N")とおく。

$t \in (T \cup T') \setminus (Ts \cup T_{in} \cup T_{out})$のとき

$\delta(\sigma_1"t, T_{in} \cup T_{out}) = \delta(\sigma_1", T_{in} \cup T_{out})$

となり、成立する。

$t \in T_{in} \cup T_{out}$のとき

$|\delta(\sigma_1'', T_{in} \cup T_{out})| = 2k$のとき、即ち、$a_{2k} = t_{k(\mathrm{mod}\ n)\text{-}in}$のとき$t = t_{k(\mathrm{mod}\ n)\text{-}out}$であることを示す。

$t = t_{i\text{-}out}$ $(i \neq k(\mathrm{mod}\ n))$とすると、$N_B$において$t_{k(\mathrm{mod}\ n)\text{-}in}$の発火後、$t_{k(\mathrm{mod}\ n)\text{-}out}$を発火せずに他の出力トランジションが発火し得ることになる。これはwell-behaved条件に反する。

$t = t_{i\text{-}in}$ $(\exists i)$とする。N''においてマーキングM_1から$t_{k(\mathrm{mod}\ n)\text{-}out}$の発火なしに、ある入力トランジションが発火可能となるならば、Nにおいて$t_{k(\mathrm{mod}\ n)}$を発火せずにトランジションt_iが発火可能となる。これは変換条件に反する。

$|\delta(\sigma_1'', T_{in} \cup T_{out})| = 2k+1$のときも同様。

補題２

$\sigma \in L(N'')$は与えられているとする。

$\phi(\sigma'') \in L(N)$, $\psi(\sigma'') \in L(N_B)$,

$M_0''[\sigma'' > M_1''$, $M_0[\phi(\sigma'') > M_1$, $M_{B0}[\psi(\sigma'') > M_{B1}$とする。このとき

$\forall p \in P$;

$$M_1(p) = M_1''(p) + \sum_{i} [OUT''(t_{i\text{-}out}, p) \{\#(\sigma'', t_{i\text{-}in}) - \#(\sigma'', t_{i\text{-}out})\}] \quad (1)$$

$$\forall p \in P'; \ M_{B1}(p) = M_1''(p) \quad (2)$$

（証明）省略。

補題３

すべての発火系列$\sigma \in L(N)$について、$\phi(\sigma'') = \sigma$となる発火系列$\sigma'' \in L(N'')$が存在する。

（証明）性質１，性質２より、発火系列の集合$L(N_B)$には、無限発火系列

$$\sigma_0\sigma_1\sigma_2\sigma_3 \cdots \sigma_{2k}\sigma_{2k+1} \cdots$$

$$\#(\sigma_{2k}, t) = \begin{cases} 1; & t = t_{k(\mathrm{mod}\ n)\text{-}in} \\ 0; & t \in (T_{in} \cup T_{out}) \setminus \{t_{k(\mathrm{mod}\ n)\text{-}in}\} \end{cases}$$

$$\#(\sigma_{2k+1}, t) = \begin{cases} 1; & t = t_{k(\mathrm{mod}\ n)\text{-}out} \\ 0; & t \in (T_{in} \cup T_{out}) \setminus \{t_{k(\mathrm{mod}\ n)\text{-}out}\} \end{cases}$$

が存在する。

性質３，N''の構成規則より、各々の発火系列$\sigma \in L(N)$においてk番目に現れるトランジション$t \in Ts$に部分系列$\sigma_k\sigma_{k+1}$を代入して得られる発火系列をσ''とすると、$\sigma'' \in L(N'')$である。また、明らかに$\phi(\sigma'') = \sigma$である。

補題４

すべての発火系列$\sigma'' \in L(N'')$について、$\phi(\sigma'') \in L(N)$である。

（証明）$\sigma'' = \lambda$のとき、$\phi(\lambda) = \lambda$より明らか。

$|\sigma_1''| = d$のとき成立すると仮定する。

$\sigma'' = \sigma_1'' t \in L(N'')$とおく。

$t \in T' \setminus T_{in}$のとき、$\phi(\sigma'') = \phi(\sigma_1'') \in L(N)$となり成立する。

$t \in (T \setminus Ts) \cup T_{in}$のとき、$\phi(\sigma'') = \phi(\sigma_1'')t \in L(N)$となることを示す。ただし$t = t_{i\text{-}in} \in T_{in}$のとき$\phi(\sigma'') = \phi(\sigma_1'')t_i$である。

補題２より

$\forall p \in P$ ； $M_1''(p) \leqq M_1(p)$

（ただし $M_0''[\sigma_1'' > M_1''$, $M_0[\phi(\sigma_1'') > M_1)$

従って、N''のM_1''において発火可能なトランジションはNのM_1においても発火可能である。よって、$\phi(\sigma'') = \phi(\sigma_1'')t \in L(N)$となる。

補題５

すべての発火系列$\sigma'' \in L(N'')$について、$\psi(\sigma'') \in L(N_B)$である。

（証明）省略。

補題６

ペトリネットＮにおいて、$Ts \subseteq T$はライブであるとする。このとき、すべての発火系列$\sigma \in L(N_B)$について、$\psi(\sigma'') = \sigma$となる$\sigma'' \in L(N'')$が存在する。

（証明）省略。

定理７（性質の保存性）

変換条件を満足するペトリネット$N = (P, T, IN, OUT, M_0)$のトランジションの集合$Ts \subseteq T$を、well-behavedなサブネット$N' = (P', T', IN', OUT', M_0')$で置き換えて得られるペトリネットを$N'' = (P'', T'', IN'', OUT'', M_0'')$とする。

(1)N''が有界であるための十分条件は、N，N_Bが有界であることである。

(2)N''が安全であるための十分条件は、N，N_Bが安全であることである。

(3)N''がライブであるための必要十分条件は、N，N_Bがライブであることである。

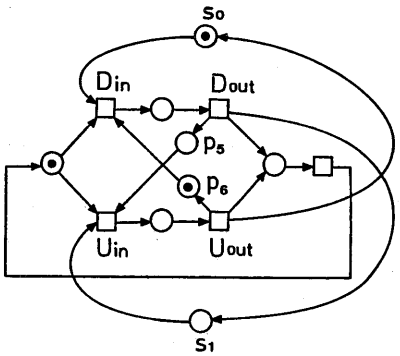
（証明）(1)(2)補題２より明らか。

(3)補題３、４、５、６より明らかである。

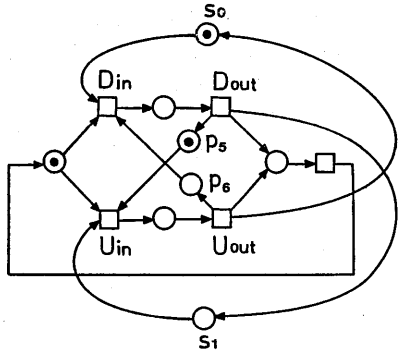
ロボットアームの例では、代置ネット図６(b)はwell-behavedでないのでライブ性は保存されない。（図８）

６．大規模システムの動作検証

著者らは現在、分散システムの動作解析用シミュレータＰＣＳＳ(Petri net-based Concurrent System Simulator)[5]を開発中である。これは、ペトリネット上で発火可能なト



(a) N_B : well-behaved



(b) N_B : not well-behaved

図８　well-behavedness

ランジションを自動的に発火させ、並行システムのシミュレーションを行うものである。

ＰＣＳＳでは、ペトリネットモデルは図９に示す階層構造をなしている。(k-1)層の各コンポーネントは、k層のあるコンポーネント中のトランジションを詳細化したサブネットである。シミュレーションは各コンポーネントを単独で実行することができ、また、必要に応じて下層のサブネットを結合し、シミュレーションすることもできる。

このような動作検証用ツールを用いてシステムを設計する際、詳細化のレベルが進むに従ってネットの規模が大きくなり、検証のための手続きも膨大になってくる。また、実際頻繁に起こるサブシステムの仕様変更に対し全体のネットのシミュレーションを必要としない検証法が望ましい。

４節に示した詳細化の条件に従ってシステムを具体化していけば各レベル間のシステムの動作が保存されるので、変更部分に対応する代置ネットの検証を行うことにより全システムの基本的性質に関する知見が得られる。

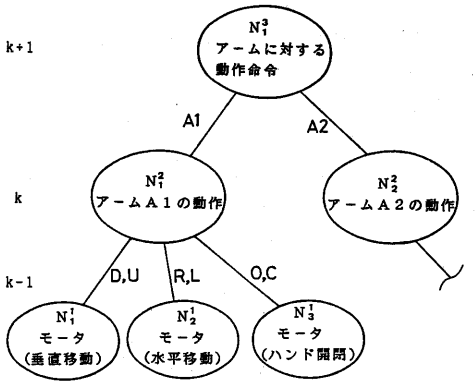


図９　ペトリネットモデルの階層構造

このように、本文で与えたシステムの階層表現法は大規模な分散システムの動作検証の効率を上げるのに貢献している。

７．むすび

本文では、ペトリネットのトランジションを他のペトリネットで置き換え順次詳細化し、システムを階層表現する際の詳細化規則を与え、その規則に従って詳細化すれば、元のネットの有界性、安全性及びライブ性等の性質が、詳細化後のネットにおいても保存されることを示した。また、このように階層表現することによって、大規模分散システムの動作検証の高効率化が図れることを示した。

文　献

[1]J.L.Peterson : " Petri Net Theory and the Modeling of the Systems,"Prentice-Hall (1981).

[2]R.Valette : "Analysis of Petri Nets by Stepwise Refinements," J. Comput. & Syst. Science, Vol.18, pp.35-46 (1979)

[3]I.Suzuki,T.Murata : "A method for stepwise refinement and abstraction of petri nets," J.Comput.& Syst. Science, Vol.27, No.1 (1983).

[4]L.Ya.Rozenblyum:"Petri Nets," Scripta Publ. Co., pp.19-43 (1984).

[5]山本,山田,内藤,辻,熊谷,児玉：" ペトリネットに基づいた並行システムシミュレータＰＣＳＳについて," 第24回SICE学術講演予稿集 (1985).