知識工学と人工知能 66 − 3
(1989. 9. 22)

ＡＴＭＳのペトリネットによる形式化について

水島 由美子　　　本位田 真一
(株) 東芝　システム・ソフトウェア技術研究所

現在エキスパートシステム等で広く使われているde KleerのＡＴＭＳをペトリネットにより形式化し、これにより得られる結果についていくつかの考察をする。ＡＴＭＳのjustificationをホーン節と同一視することにより、ＡＴＭＳにLautenbach's netというペトリネットを対応させる。すると、ＡＴＭＳのノードのラベルを線形方程式を用いて求めることができる。また、この方法でノードのラベルを求めると、ラベルと同時にラベルの各々の環境が、どのjustificationを用いて導くことができるかを知ることが可能となる。これにより、ラベルがどのようにして導かれたかに答える事や、basicなＡＴＭＳでは容易でないとされていたjustificationの削除が可能になった。

Using a Petri net to model and analyze ATMS

Yumiko Mizushima　　Shinichi Honiden

Systems & Software Engineering Laboratory ,TOSHIBA Corporation

70 Yanagicho ,Saiwaiku, Kawasaki 210 ,Japan

De Kleer's ATMS is extensively used for expert systems, and known as one of the best frame to formalize the assumption based reasoning. In this paper, we formalize ATMS by using a Petri net and discuss the results obtained from this formalization. A justification of ATMS identifies a Horn clause, hence ATMS corresponds to a Petri net called Lautenbach's net. From this correspondence, we obtain the label for each node by solving linear equations for its roots. Using this method, we know a set of justifications leading to the environment for the label, and it becomes possible to remove justifications.

1．はじめに

De KleerのＡＴＭＳ[1]は、推論や問題解決を行うシステム（以後、対象システムとよぶ）に付随して対象システムの持つ信念の系を整合的に管理するシステムである。ＡＴＭＳは仮説推論を現在のところ最も適切に定式化した枠組みであり、エキスパートシステム等に広く使われているが、仮説の組合せ爆発の問題、計算コストやメモリ消費が大きいこと等が指摘されている。この問題は、エキスパートシステムが扱うノード数が多くない場合はさほど性能低下の原因にならないが、知識ベースが大規模化（ノードの数が数万～数十万）した際には顕著になるのは自明である。そのため、ＡＴＭＳの並列化の必要性が明らかにされており[2]、例えばコネクションマシン上での実装に関する研究が行われている[3]。

そのような流れの中で著者らは、身近なアーキテクチャにおいて種々の高速化手法が確立されている成果を積極的に利用する事も有効であると考える。その成果の一つとして大型行列計算におけるスーパーコンピュータの利用がある[4]。この成果を利用するためには、ＡＴＭＳにおけるラベル計算を何らかの手段によって行列計算の世界に持ち込む必要がある。そこで、ここで用いる手法としては、まずホーン節の集合には Lautenbach's net[5]というペトリネットが対応する事に着目し、ＡＴＭＳのjustificationをホーン節と同一視し、ＡＴＭＳに対するペトリネットを定義する。次にLautenbach's netに関して文献[6]に有効な結果が存在する事を利用し、ラベルを〝Lautenbach's netに対する連立一次方程式の解〟という問題に帰着させる方法について述べる。

また、従来の方法でラベルを求める時、解の探索過程を直接見る事ができなかったため、対象システムがＡＴＭＳの探索を制御する事ができない事、justificationの削除が困難である事等の問題点もあった。本手法を用いると、ラベルと同時にラベルの各環境を導くために用いたjustificationを知ることができ、上記の問題をも克服する事が可能であるという利点を持つ。

2．ＡＴＭＳ

ＡＴＭＳは予め与えられた仮説をもとに推論を行なうために、データベースの無矛盾性や推論した結果を管理するシステムである。簡単にＡＴＭＳで用いる用語の説明をするが、ここでは本稿で用いる最低限の用語についてのみ記すことにする。ＡＴＭＳについて詳しくは、文献[1]を参照されたい。

(1)．ノード

ノードとは対象システムの個々のデータに対応するＡＴＭＳのデータをいう。自分自身のみを理由づけに持つノードを仮説ノードという。仮説ノードは推論の基準となるデータである。また矛盾である事を表すノードを矛盾ノードといい⊥であらわす。

(2)．justification

justificationとはあるノードが、他のノードからどのようにして導かれたかを表現するもので、$b_1 \wedge \cdots \wedge b_n \rightarrow c$の形をとる。（ここで$b_i$は矛盾ノード以外のノード，cは仮説ノード以外のノードである）これは、$b_1, \cdots, b_n (i \leqq i \leqq n)$が成立するならばcが成立することを意味する。また、b_iをantecedentノードと言いcをconsequentノードと言う。

(3)．環境

仮説ノードから成る集合を環境という。

(4). ラベル

ラベルとは環境の集合で、ノードがどの仮説ノードに基づくかを示すものをいう。例えばノードnのラベルが $\{\{a_1\},\{a_2,a_3\}\}$ であるとは、a_1が成立するか、またはa_2とa_3が成立するときに、nが成立するという事を表す。ラベルはconsistent（ラベルに含まれる各環境が無矛盾である），sound（ノードがラベルの各環境から導かれる），complete（環境Ｅがノードnを導くならば、nのラベルはＥ⊃Ｅ'なる環境Ｅ'を含む），minimal（ラベルの各環境はラベルの他の環境のスーパーセットではない）という四つの性質を満たすものとする。

(5). nogood

矛盾を起こす環境をnogoodという。⊥のラベルに含まれる環境のスーパーセットとなる環境がnogoodである。

(6).コンテキスト

環境Ｅに対するコンテキストとは、Ｅを仮定すると導くことができるノードの集合をいう。

ＡＴＭＳの主な仕事は、justificationが新しく加わった時に、ラベルが上記の四つの性質を満たすようにラベルの更新を行うことである。ラベルの更新のアルゴリズムについては、文献[1]を参照されたい。

３．ペトリネット

<u>定義</u>

ペトリネット構造とは５項組（Ｐ，Ｔ，Ｑ，M_0，Ｗ）をいう。ここでＰ，Ｔ，Ｑ，M_0，Ｗは次のものである。

Ｐ:プレースの有限集合

Ｔ:トランジションの有限集合

Ｑ:プレースからトランジションへのアーク及びプレースからトランジションへのアークの有限集合

M_0:初期マーキング　　$M_0 : P \longrightarrow \{0,1,2,\cdots\}$　(mapping)

Ｗ:重み（アークの本数）　　$W : Q \longrightarrow \{1,2,3,\cdots\}$　(mapping)

<u>定義</u>

トランジション t が goalであるとはtから出て行くアークが存在しないことをいう。

<u>定義</u>

ペトリネットに対する合成推移行列とはn×m 行列$B=(b_{ij})$をいう。ここで nはトランジションの数mはプレースの数である。各成分b_{ij}とは〝トランジションiからプレースjへのアークの重み〞から〝プレースjからトランジションiへのアークの重み〞を引いたものをいう。

<u>定義</u>[5]

Ｒを命題論理式の集合とする。Ｒに対するLautenbach's net （Ｐ，Ｔ，Ｑ，M_0，Ｗ）とは次のものをいう。

(a).Ｔ＝Ｒ

(b).Ｐ＝Ｒに現れるアトム

(c).$Q \subseteq P \times T \cup T \times P$

ただし$(p,t)\in Q\cap(P\times T)\Leftrightarrow$not pがtに現れる　，$(t,p)\in Q\cap(T\times P)\Leftrightarrow$pがtに現れる
(d).初期マーキング $M_0=0$
(e).重みは１

命題（G.Peterka and T.Murata,[6]）
Ｒを有限個のホーン節からなる集合、(P,T,Q,M_0,W)をＲに対応するLautenbach's netとする。Ｎの合成推移行列をＢ、Ｂの転置行列を${}^tB$とする。Ｒがただ一つのgoal節gを含むとし、gに対するトランジションをt_gとする。このとき次の(i),(ii)は同値である。
(i).Ｒは矛盾を含む
(ii).${}^tB\vec{x}=\vec{0}$, $\vec{x}\geqq 0$なる整数成分のベクトル$\vec{x}$でt_gに対する成分が0でないものが存在する
（ここで$\vec{x}\geqq 0$は$\vec{x}$の各成分が0以上である事を表す。）

注意
Ｒに矛盾があったとする。この時Ｒに矛盾がある事を導くのにホーン節ｈを用いる事と、$\vec{x}$のｈに対する成分が正である事とは同値である。

４．ＡＴＭＳのペトリネットによる形式化

(1).記号
以下、次の記号を下記のものとして扱う事にする。
$\mathbf{N}=\{0,1,2,\cdots\}$
$A=\{a_1,\cdots,a_m\}$：仮説ノードの集合
$N=\{a_1,\cdots,a_m,b_1,\cdots,b_n\}$：ノードの集合
$J=\{j_1,\cdots,j_k\}$：justificationの集合
$\vec{e_i}$：i番目が1でそれ以外が0の(m+n)次元縦ベクトル
$\sigma(a_i)=\vec{e_i}$ $(1\leqq i\leqq m)$と置き、これをa_iに対するベクトルという
$\sigma(b_t)=\vec{e_{m+t}}$ $(1\leqq t\leqq n)$と置き、これをb_tに対するベクトルという
justification　$j:b_1'\wedge b_2'\wedge\cdots\wedge b_t'\rightarrow c'$に対して　$\sigma(j)=\sum_{i=1}^{t}\sigma(b_i')-\sigma(c')$と置く。これをjに対応するベクトルという。

(2).ＡＴＭＳのペトリネットによる形式化
ＡＴＭＳのjustification　$b_1'\wedge\cdots\wedge b_t'\rightarrow c$ をホーン節 $b_1'\wedge\cdots\wedge b_t'\supset c$ と同一視する。これによりＡＴＭＳにjustificationの集合ＪのLautenbach's netが対応する。

定義
ＡＴＭＳ(A,N,J)に対するペトリネット(P,T,Q,M_0,W)を次のように定義する。
(a).$T=A\cup J$

(b). P＝N

(c). $Q \subseteq P \times T \cup T \times P$

ただし、$(p,t) \in Q \cap (P \times T)$ ⇔pがtのantecedentノードとして現れる

$(t,p) \in Q \cap (T \times P)$ ⇔pがtのconsequentノードとして現れる

(d). 初期マーキング $M_0=0$

(e). 重みは１

<u>定義</u>

ATMS（A, N, J）に対するペトリネットに、次の$(m+n)\times(m+k)$次元行列Sを対応させ、これをATMS（A, N, J）に対する行列と定義する。

$$S=[\sigma(a_1),\cdots,\sigma(a_m),\ \sigma(j_1),\cdots,\sigma(j_k)]$$

<u>定理１</u>

Eを環境, bをノードとする。Sに$-\sigma(b)$を付け加えた$(m+n)\times(m+k+1)$次元行列を$M_b=[S,-\sigma(b)]$と置く。即ち$M_b=[\sigma(a_1),\cdots,\sigma(a_m),\ \sigma(j_1),\cdots,\sigma(j_k),-\sigma(b)]$．すると、次の(i), (ii)は同値である

(i). Eからbを導く事ができる

(ii). $M_b{}^t\vec{x}=\vec{0}, \vec{x}\geqq 0$なる$\vec{x}$で、$-\sigma(b)$に対する成分が正であるものが存在する

（証明）

EからJを用いてbを導ける事と、$E \cup J \cup \{\sim b\}$ に矛盾がある事とは同値である。従って命題より定理１を得る。■

5. 諸結果

(1). ラベル

<u>定義</u>

$\mathbf{N}^r=\mathbf{N}\times\cdots\times\mathbf{N}$において元の順序を次のように入れる。

$\mathbf{N}^r \ni \vec{x}=(x_1,\cdots,x_r), \vec{y}=(y_1,\cdots,y_r)$に対して$\vec{x} \leqq \vec{y}$ および $\vec{x} < \vec{y}$ を次のように定義する。

$\vec{x} \leqq \vec{y}$ ⇔ 任意の$i(1\leqq i\leqq r)$に対して$x_i \leqq y_i$

$\vec{x} < \vec{y}$ ⇔ $\vec{x} \leqq \vec{y}$であり、かつある$j(1\leqq j\leqq r)$が存在して$x_j<y_j$

<u>定理２</u>

(P, T, Q, M_0, W) をATMSに対応するペトリネット、M_bを上記の$(m+n)\times(m+k+1)$行列とする。集合$V_b=\{\vec{x}=(x_1,\cdots,x_{m+k+1})\in\mathbf{N}^{m+k+1} \mid M_b{}^t\vec{x}=\vec{0},\ x_{m+k+1}>0\}$ の極小元の集合を $\{\vec{y}_1,\cdots,\vec{y}_t\}$ とする。

$\vec{y}_i=(x_{i1},\cdots,x_{i(m+k)},x_{i(m+k+1)})$とし、$\vec{x}_i=(x_{i1},\cdots,x_{i(m+k)})$，$K_b=\{\vec{x}_1,\cdots,\vec{x}_t\}$ とする。各$\vec{x}_i\in K_b$に対して$E_i=\{a_j\in A \mid x_{ij}\neq 0\}$と置く。集合 $\{E_1,\cdots,E_t\}$ の包含関係に関する極小元

全体の集合をL_bとすると、L_bはノードbのsound,complete,minimalなラベルである。

（証明）

ラベルL_bがsoundであること、すなわち環境E_iがノードbを導くことを示す。$\vec{x}_i \in K_b$に対して

$x_{i(m+k+1)}$を式 $x_{i(m+k+1)}\sigma(b)=\sum_{u=1}^{m} x_{iu}\sigma(a_u)+\sum_{v=1}^{k} x_{i(m+v)}\sigma(j_v)$により定まる自然数とする。

$(\vec{x}_i, x_{i(m+k+1)})$は$V_b$の元だから定理１の(ii)の条件を満たす。従って、定理１よりE_iからbを導くことができる。

ラベルL_bがcompleteである事、即ち環境Eがbを導くなら$E \supset E_i$なるi($1 \leqq i \leqq t$)が存在する事を示す。ある環境Eがノードbを導いたとすると、定理１より $M_b{}^t\vec{y}=\vec{0}$，$y_{m+k+1}>0$を満たす

$\vec{y}=(y_1,\cdots,y_{m+k+1})$ ($\in \mathbf{N}^{m+k+1}$) が存在する $(\vec{x}_1, x_{1(m+k+1)}),\cdots,(\vec{x}_t, x_{t(m+k+1)})$は$V_b$の極小元の全てだから、この$\vec{y}$はある$(\vec{x}_i, x_{i(m+k+1)})$を用いて$\vec{y}=(\vec{x}_i, x_{i(m+k+1)})+\vec{y}'$ ($\vec{y}' \in \mathbf{N}^{m+k+1}$)と書ける。命題の所で注意した事から$y_j$が正である事と$a_j \in E$である事とは同値である。従って

$\vec{y}' \in \mathbf{N}^{m+k+1}$より$E_i \subset E$を得る。

また、L_bの元は$\{E_1,\cdots,E_t\}$の包含関係に関する極小元なのでL_bの元は他の元のスーパーセットになる事はない。従ってL_bはminimalである。■

定理２で求めたラベルL_bから⊥のラベルの環境のスーパーセットになっている環境を削除したものを求め、これをL_b'とすればL_b'はbのsound,complete,minimal,consistentなラベルである。

本手法では、ラベルとして定理２のK_bを保管し、対象システムから″ノードbのsound,complete,minimal,consistentなラベルを求めたい″という要請があった時に、K_bよりL_bを求め、L_bより⊥のラベルの環境のスーパーセットになっている環境を削除したL_b'を求めることにする。これによるメリットとしては、nogoodを削除してもラベルに対する影響がない事、およびラベルを求める際のjustificationの適用する過程が明示される事、（ラベルの更新については以下に記すが）ラベルの更新が容易になる事が挙げられる。

実際に極小元を求める事については、線形計画法などの有益な結果を利用する。

(2).コンテキスト

<u>定理３</u>

(P,T,Q,M_0,W) をATMSに対応するペトリネット、M_bを上記の(m+n)×(m+k+1)行列、Eを無矛盾な環境とする。この時Eに対するコンテキストは

$X_E=\{b \in N \mid \sigma(b) \in \sum_{i=1}^{m} \mathbf{N}\sigma(a_i)+\sum_{t=1}^{k} \mathbf{N}\sigma(j_t)\}$ で与えらる。

（証明）

$M_b=[\sigma(a_1),\cdots,\sigma(a_m)\ \sigma(j_1),\cdots,\sigma(j_k),-\sigma(b)]$ と置く。ノードbに対して

$\sigma(b)=\sum_{1\leqq i\leqq m}\lambda_i\sigma(a_i)\ +\ \sum_{1\leqq t\leqq n}\mu_t\sigma(j_t)$なる$\lambda_i, \mu_t \in \mathbf{N}$が存在すると仮定する。行列 M_bとベクトル$\vec{y}=(\lambda_1,\cdots\lambda_n,\mu_1,\cdots,\mu_k,1)$について、$M_b{}^t\vec{y}=0$ が成立する。${}^t\vec{y}$は定理１の(ii)の条件を満たすから、定理１よりEからbを導くことができる。従ってbはEのコンテキストの元である。

逆にbをEのコンテキストの元とする。Eよりbを導くのに用いるjustificationの個数の最小値を$\ell_E(b)$で表す。この時に用いるjustificationでbをconsequentノードに持つものを$j_b:c_1\wedge\cdots\wedge c_r\to b$とする。この時、$\ell_E(c_t)<\ell_E(b)(1\leqq t\leqq r)$である事に注意する。

$\sigma(b)\in\sum_{a\in A}\mathbf{N}\sigma(a_i)+\sum_{j\in J}\mathbf{N}\sigma(j)$ (*)である事を$\ell_E(b)$に関する数学的帰納法により示す。$\ell_E(b)=1$の時、$\sigma(b)=\sum_{a\in A}\lambda(a)\sigma(a)$ $(\lambda(a)=0 \text{ or } 1)$より(*)が成立。$\ell=\ell_E(b)$とし、$\ell$より小さい自然数に対して(*)が正しいと仮定した時ℓについても(*)が正しい事を以下に示す。$\ell_E(c_t)<\ell_E(b)$ゆえ

$\sigma(c_t)=\sum_{a\in A}\lambda_t(a)\sigma(a)+\sum_{j\in J}\mu_t(j)\sigma(j)$ なる$\lambda_t(a),\mu_t(j)\in\mathbf{N}(i\leqq t\leqq r)$が存在する。

$\sigma(j_b)=-\sum_{t=1}^{r}\sigma(c_t)+\sigma(b)$ ゆえ

$$\sigma(b)=\sigma(j_b)+\sum_{t=1}^{r}\sigma(c_t)=\sigma(j_b)+\sum_{a\in A}\Big(\sum_{t=1}^{r}\lambda_t(a)\Big)\sigma(a)+\sum_{j\in J}\Big(\sum_{t=1}^{r}\mu_t(j)\Big)\sigma(j)$$

となり、ℓについても(*)が成立する。■

6．ラベル更新

(1).各ノードに対する更新

ラベルの更新のために、各ノードに対して定理２における極小解より得られた集合K_b $(=K_{J,b})$ は保持しておく。

以下の定理４，５において次の写像により$\mathbf{N}^n$を$\mathbf{N}^{n+1}$の部分集合と見なす事にする。

$$\begin{array}{ccc}\mathbf{N}^n & \longrightarrow & \mathbf{N}^{n+1}\\ \cup\!\!\!\!\!\!\;\text{ι} & & \cup\!\!\!\!\!\!\;\text{ι}\\ (x_1,\cdots,x_n) & \longmapsto & (x_1,\cdots,x_n,0)\end{array}$$

<u>定理４</u>（justificationの追加によるラベルの更新）

$b_1'\wedge b_2'\wedge\cdots\wedge b_t'\to c'$なるjustificationを新たにＪに付け加えるものとする。$c'=b\in N$のとき$K_{J\cup\{j\},b}=K_{J,b}\oplus K'$ である。ここでK'は次のようにして求まる集合である。

$\{\vec{x}=(x_1,\cdots,x_{m+k+2})\in\mathbf{N}^{m+k+2}\mid M_{J\cup\{j\},b}{}^t\vec{x}=\vec{0},x_{m+k+1}>0,x_{m+k+2}>0\}$ の極小元の集合を$\{\vec{y}_1,\cdots,\vec{y}_t\}$ とし、$\vec{y}_i=(x_{i1},\cdots,x_{i(m+k+1)},x_{i(m+k+2)})$とするとき、

$\vec{z}_i=(x_{i1},\cdots,x_{i(m+k+1)})$と置けば$K'=\{\vec{z}_1,\cdots,\vec{z}_t\}$．ただし、$\oplus$は集合の直和を表す。

（証明）

$$\begin{aligned}&\{\vec{x}=(x_1,\cdots,x_{m+k+2})\in\mathbf{N}^{m+k+2}\mid M_{J\cup\{j\},b}{}^t\vec{x}=\vec{0},x_{m+k+2}>0\}\\ =&\{\vec{x}=(x_1,\cdots,x_{m+k+2})\in\mathbf{N}^{m+k+2}\mid M_{J\cup\{j\},b}{}^t\vec{x}=\vec{0},x_{m+k+1}=0,x_{m+k+2}>0\}\\ \oplus&\{\vec{x}=(x_1,\cdots,x_{m+k+2})\in\mathbf{N}^{m+k+2}\mid M_{J\cup\{j\},b}{}^t\vec{x}=\vec{0},x_{m+k+1}>0,x_{m+k+2}>0\}\end{aligned}$$

より定理４を得る。■

注意　定理４のc'が$c'\notin N$である時は定理２により$K_{J\cup\{j\},c'}$を求める。

定理５（justificationの削除によるラベルの更新）

j_v: $b_1' \wedge b_2' \wedge \cdots \wedge b_t' \to c'$ なるjustificationをＪから削除するものとする。この時

$K_{J-\{j_v\}} = \{\vec{z} \in K_J \mid z_{m+v}=0\}$ である。

（証明）

$K_{J-\{j_v\}} \supset \{\vec{z} \in K_J \mid z_{m+v}=0\}$ である事は命題の所で注意した事により明か。

$K_{J-\{j_v\}} \subset \{z \in K_J \mid z_{m+v}=0\}$ である事を示す。

$K_{J-\{j_v\}}$は$\{\vec{x}=(x_1,\cdots,x_{m+k+1}) \in \mathbf{N}^{m+k+1} \mid M_J{}^t\vec{x}=\vec{0},\ x_{m+k+1}>0, x_{m+v}=0\}$ の極小元の集合の$\mathbf{N}^{m+k}$への正射影である。よって $K_{J-\{j_v\}}$の任意の元$\vec{z}$に対しK_Jの元$\vec{x}$が存在し、$(\vec{z}, z_{m+k+1}) \geq (\vec{x}, x_{m+k+1})$を満たす。故に $0=z_{m+v} \geq x_{m+v} \geq 0$より$x_{m+v}=0$となり$\vec{x} \in K_{J-\{j_v\}}$を得る。$\vec{z}$の極小性より$\vec{z}=\vec{x}$であるから$\vec{z} \in K_J$である。■

(2). ラベル更新の順序

justificationの集合Ｊに変化があった場合、どのノードのラベルをどの順序で更新するべきかについてはdependencyグラフを利用する。

７．例

EX.1

ノードをa1,a2,a3,b,c,dとし、仮説ノードをa1,a2,a3とする。また次のj1～j5をjustificationとする。

j1: $a1 \wedge a2 \supset b$　j2: $a2 \wedge a3 \supset c$

j3: $a3 \wedge b \supset d$　j4: $c \supset d$　j5: $a1 \wedge a2 \supset \perp$

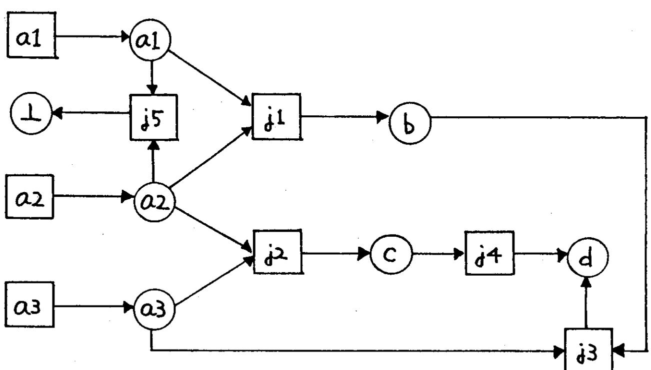
このＡＴＭＳに対応するペトリネットを（図１）に、対応する行列を（図２）に示す。また、以上のjustificationを用いて従来の方法で各ノードのラベルを計算した結果を（表１）に示す。

$$
S = \begin{array}{c|rrrrrrrr|}
 & a1 & a2 & a3 & j1 & j2 & j3 & j4 & j5 \\
a1 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 \\
a2 & 0 & 1 & 0 & -1 & -1 & 0 & 0 & -1 \\
a3 & 0 & 0 & 1 & 0 & -1 & -1 & 0 & 0 \\
b & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\
c & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\
d & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
\perp & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}
$$

（図２）

（表１）

ノード	ラベル
b	{ }
c	{{a2, a3}}
d	{{a2, a3}}
⊥	{{a1, a2}}



（図１）

ここでは本手法を用いてノードdのsound,complete,minimal,consistentなラベルを求める。

まず、⊥のラベルを求める。

$M_\perp=[S,-\sigma(\perp)]$，$V_\perp=\{\vec{x}=(x_1,\cdots,x_9)\in\mathbf{N}^9 \mid M_\perp{}^t\vec{x}=\vec{0}$ かつ $x_9>0\}$ と置く。$\vec{x}$（$\in V_\perp$）は ${}^t x=\lambda(1,1,0,0,0,0,0,1,1)$（ここでλは自然数）なる形をしている。これより$V_\perp$の極小元の集合は $\{(1,1,0,0,0,0,0,1,1)\}$ 従って、$K_\perp=\{(1,1,0,0,0,0,0,1)\}$ である。これの成分が正の所に対応する仮説ノードはa1,a2、justificationはj5であるから、定理２より⊥のラベルはa1∧a2で、これを導くのにj5が必要であることがわかる。

次にdのsound,complete,minimalなラベルを求める。

$M_d=[S,-\sigma(d)]$，$V_d=\{\vec{x}=(x_1,\cdots,x_9)\in\mathbf{N}^9 \mid M_d{}^t\vec{x}=\vec{0}$ かつ $x_9>0\}$ と置く。$\vec{x}$（$\in V_d$）とする時${}^t\vec{x}$は次の形をしている。$\lambda(1,0,0,1,-1,1,-1,0,0)+\mu(0,1,1,0,1,0,1,0,1)$　（ここでμは自然数，$\lambda\in N$）。これよりV_dの極小元集合は $\{(0,1,1,0,1,0,1,0,1),(1,1,1,1,0,1,0,0,1)\}$ であり、$K_d=\{(0,1,1,0,1,0,1,0),(1,1,1,1,0,1,0,0)\}$ である。集合 {{a2,a3},{a1,a2,a3}} の包含関係に関する極小元は{a2,a3}であるから$L_d=\{\{a2,a3\}\}$、従って定理２よりdのsound,complete,minimalなラベルはa2∧a3である事がわかる。またK_dの要素(0,1,1,0,1,0,1,0)を参照する事により、a2∧a3を求めるのにj2とj4を用いたことがわかる。

a2∧a3は、⊥のラベルの要素a1∧a2のスーパーセットになっていないからa2∧a3はconsistentである。

ここでdのラベルを求めるのに用いたjustificationはj2とj4のみであるから、これ以外のjustificationを削除してもdのラベルは変化しない。justification j2を削除したとする。

$V_{J-\{j2\},d}=\{\vec{x}=(x_1,\cdots,x_9)\in\mathbf{N}^9 \mid M_b{}^t\vec{x}=\vec{0}$ かつ $x_9>0,\ x_5=0\}$ について、$K_{J-\{j2\},d}$は $\{(1,1,1,1,0,1,0,0)\}$ であるから、a1∧a2∧a3を仮定すればj1,j3を用いてdを導くことが可能である。一方a1∧a2∧a3は矛盾ノードのラベルの要素のスーパーセットであるから結局j2を削除するとdのラベルは空となる。j4を削除した場合も同様である。

EX.2

ノードをa1,a2,a3,b,c,d、仮説ノードをa1,a2,a3とする。また次のj1～j7をjustificationとする。

j1: a1⊃b　j2: a2⊃c　j3: a3⊃d　j4: b⊃c　j5: c⊃b　j6: c⊃d　j7: d⊃c

以上のjustificationを用いてbのノードのラベルを従来の方法で計算すると {{a1},{a2},{a3}} となる。

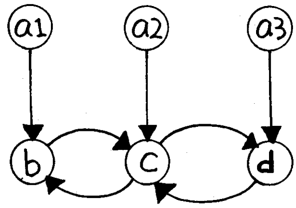
$V_b=\{\vec{x}=(x_1,\cdots,x_{11})\in\mathbf{N}^{11} \mid M_b{}^t\vec{x}=\vec{0},x_{11}>0\}$ の元は下記の形をしている。

$$\vec{x}=\alpha(1,0,0,1,0,0,\ 0,0,0,0,1)+\beta(0,1,0,0,1,0,-1,0,0,0,1)+\gamma(0,0,1,0,0,1,-1,0,0,1,1)$$
$$+\delta(0,0,0,0,0,0,\ 1,1,0,0,0)+\varepsilon(0,0,0,0,0,0,\ 0,0,1,1,0)$$

（ここで、$\alpha,\beta,\gamma,\delta,\varepsilon\in\mathbf{N}$、α，β，γのうちの少なくとも一つは正）

V_bの極小元の集合は $\{(1,0,0,1,0,0,0,0,0,0,1),(0,1,0,0,1,0,0,1,0,0,1),(0,0,1,0,0,1,0,1,0,1,1)\}$ であるから、$K_b=\{(1,0,0,1,0,0,0,0,0,0),(0,1,0,0,1,0,0,1,0,0),(0,0,1,0,0,1,0,1,0,1)\}$ となる。従って、bのラベルは {{a1},{a2},{a3}} であり{a1}を求めるのにはj1を、{a2}を求めるのにはj2,j5をを、{a3}を求めるのにはj3,j5,j7を用いた事がわかる。

この例のdependencyグラフを（図３）に示すが、このようにノード間にループが存在し関係が複雑な時、従来の方法では一つのノードに対するラベルの更新は一度では済まされない場合が多かった。しかし上に示したように本手法を用いればそれぞれのノードのラベルの計算は一度で済むという利点を持つ。



（図３）

８．おわりに

本稿ではＡＴＭＳの並列処理化を目的として、ＡＴＭＳをペトリネットにより形式化する方法について述べた。また、形式化する事によりラベルを求める事を線形方程式の解を求める事に帰着することが可能であることを示した。本手法によるスーパーコンピュータ上でのＡＴＭＳの高速な実装方式以外にも、コネクションマシン上での実装に関する方法があるが、これらとの比較について実装評価結果をふまえて別途報告する予定である。

更に、本手法ではラベルと同時にラベルの各環境を導くのに必要なjustificationも求める事ができる。この結果として、従来のＡＴＭＳでは容易ではないとされていたjustificationの削除、ラベルがどのようにして導かれたかを知る事が可能となった。現在ＡＴＭＳはエキスパートシステムに広く使われているが、これはその一つの機能としての説明機能を補助する手段としても有効である。

参考文献

[1]. de Kleer, J., An Assumption-based TMS, Artificial Intelligence 28, pp.127～162, 1986.

[2].de Kleer, J.,Problem Solving with the ATMS, Artificial Intelligence 28, pp.197～224, 1986.

[3]. Dixon, M. and de Kleer, J., Massively Parallel Assumption-based Truth Maintenance,in Proc. of AAAI-88,1, pp.199-204.

[4].名取 亮 野寺 隆，大規模行列計算における反復解法，情報処理，vol.28, no.11, pp.1452-1459,Nov.1987.

[5]. K.Lautenbach, On logical and linear dependencies, Sankt Augustin, Germany, GMD Rep.147,1985.

[6]. G.Peterka and T.Murata, Proof Procedure and Answer Extraction in Petri Net Model of Logic Programs, IEEE Tràns.on Software Eng.,vol.15, no.2, pp.209-217, Feb.1989.