Title	計算モデルとしての推論加群系 (アルゴリズムと計算の理論)
Author(s)	山崎, 勇
Citation	数理解析研究所講究録 (1998), 1041: 111-118
Issue Date	1998-04
URL	http://hdl.handle.net/2433/62064
Right	
Type	Departmental Bulletin Paper
Textversion	publisher

数理解析研究所講究録
1041 巻 1998 年 111-118

計算モデルとしての推論加群系

山崎　勇
Isamu YAMAZAKI
東芝　研究開発センター　情報・通信システム研究所

1998 年 2 月 3 日

1　はじめに

筆者はこれまでに，論理を指向した代数系として推論加群系を定義し，その上で論理の問題を表現できることを示した [5, 6, 7]．例えば「証明問題は証明方程式に非負の解があるかどうかという問題に変換できる」という代数的証明原理を導いた．

ところでこの推論加群系における環 R と M の元は，項を加工する操作を表現していると見ることができる．すなわち「計算」を「項の変形」に限れば，これらの環の元は計算能力を持つ．そこで計算を記述・実行するためのモデル (計算モデル) として，これらの環をみたときに，その能力 (計算能力) がどの程度であるかということが問題となる．そこで環 R の元の計算能力について分析を行った．

その結論として，無限級数にまで拡大した推論加群系では，環 R の元の計算能力は，チューリングマシンや一般帰納的関数と同等であることを得た．その結果，例えば R における無限級数の収束の判定は決定不能であることが結論される．これはチューリングマシンの停止性判定問題が無限級数の収束性判定問題に変換できることによる．さらに，多重の無限級数を 1 重の無限級数に変換できることを見出した．本稿ではこれらについて報告する．

2　推論加群系

この節では，以後の節で使用する推論加群系についての概要を述べる．詳しくは [7] を御参照いただきたい．

◆諸定義　項の全体を H と記す．基礎項の全体を H_T と記す．項の列を $\hat{t}$, $\hat{s}$ などと記す．変数記号の列を $\hat{X}$, $\hat{Y}$, $\hat{X}_k$ などと記す．項，項の列，素論理式を**表現**と称し，E と記す．変数記号を含む表現を $E[X,Y]$, $E[\hat{X}]$, $\hat{t}[\hat{Y}]$ などと記す．表現 E が含む変数記号を，E の**代表変数**と称し，その全体を $\boldsymbol{Y}(E)$ と記す．述語記号が P, Q, P_k などである時，素論理式を $P(\hat{t})$, $Q(\hat{X})$, $P_k(\hat{s})$ などと記す．素論理式の全体を G と記す．基礎式 (基礎素論理式) の全体を G_T とする．$\tilde{P}$, $\tilde{Q}$, $\tilde{P}_k$ などを**事実記号**と称する．$\tilde{P}(\hat{t})$, $\tilde{Q}(\hat{X})$, $\tilde{P}_k(\hat{s})$ などを**素事実**と称し，その全体を $\tilde{G}$ と記す．

対象とする問題には λ(有限) 個の述語記号 P_k $(k=1,\cdots,\lambda)$ が現れるものとする．またこれとは別に arity=0 の基準述語記号 P_0 を考える．$K_0=\{0,1,2,\cdots,\lambda\}$ とする．特に断らない限り $\sum$ は実質的有限和を表すものとする．

◆総代入　基礎項 t_0 を任意に選んで固定する．次により**総代入**と呼ぶ表現から表現への右写像 $\left\langle\frac{\hat{t}}{\hat{X}}\right\rangle$ を定義する．

$$E[\hat{X},\hat{Y}]\left\langle\frac{\hat{t}}{\hat{X}}\right\rangle=E[\hat{t},\hat{u}] \qquad (\,\hat{u}=(t_0,t_0,t_0,\cdots)\,)$$

すなわち，総代入 $\left\langle\frac{\hat{t}}{\hat{X}}\right\rangle=\left\langle\frac{t_1}{X_1},\frac{t_2}{X_2},\cdots,\frac{t_n}{X_n}\right\rangle$ は，表現中の変数記号のうち，$X_i\in\{\hat{X}\}$ は項 $t_i\in\hat{t}$ で置き換え，$\hat{X}$ に含まれない変数記号は t_0 で置き換える機能を持つ．総代入の全体を T_X と記す．$\hat{t}$ が基礎項のみからなる総代入を**基礎総代入**と称し，その全体を T と記す．

総代入 $\theta=\left\langle\frac{\hat{t}}{\hat{X}}\right\rangle$ において $\boldsymbol{X}(\theta)\overset{\mathrm{def}}{=}\{\hat{X}\}$ に含まれる変数記号を θ の**指定変数**と称する．また $\boldsymbol{Y}(\theta)\overset{\mathrm{def}}{=}\boldsymbol{Y}(\hat{t})$ に含まれる変数記号を θ の**代表変数**と称する．

◆基本加群系　まず推論加群系の土台となる基本加群系を定義する．

有理数の全体 $\boldsymbol{Q}$ の上で T から生成する右 $\boldsymbol{Q}$ 加群を R_T, $\boldsymbol{Q}$ の上で G_T から生成する右 $\boldsymbol{Q}$ 加群を D_T とする．

$$\mathrm{R}_T\overset{\mathrm{def}}{=}\left\{\sum_j\rho_j\cdot r_j\;\middle|\;\rho_j\in T,\ r_j\in\boldsymbol{Q}\right\}$$
$$\mathrm{D}_T\overset{\mathrm{def}}{=}\left\{\sum_j A_j\cdot r_j\;\middle|\;A_j\in G_T,\ r_j\in\boldsymbol{Q}\right\}$$

$r\in\boldsymbol{Q}$ の $\beta\in\mathrm{R}_T$ および $d\in\mathrm{D}_T$ への右作用をそれぞれ $\beta\cdot r$, $d\cdot r$ と記す．

次に R_T から R_T への準同型写像の全体を $\mathcal{R}$, R_T から D_T への準同型写像の全体を $\mathcal{D}$, D_T から R_T への準同型

写像の全体を $\mathcal{P}$ と定義する．

$$\mathcal{R} = \mathrm{Hom}_Q(\mathsf{R}_T, \mathsf{R}_T),$$
$$\mathcal{P} = \mathrm{Hom}_Q(\mathsf{D}_T, \mathsf{R}_T), \qquad \mathcal{D} = \mathrm{Hom}_Q(\mathsf{R}_T, \mathsf{D}_T)$$

$\alpha \in \mathcal{R}$ による $\beta \in \mathsf{R}_T$ の像を $\alpha \cdot \beta$ と記す．$d \in \mathcal{D}$ による $\beta \in \mathsf{R}_T$ の像を $d \cdot \beta$ と記す．$\psi \in \mathcal{P}$ による $d \in \mathsf{D}_T$ の像を $\psi * d$ と記す．また，$\alpha \in \mathcal{R}$ と $\beta \in \mathcal{R}$ との合成写像を $\beta \cdot \alpha$ $(\in \mathcal{R})$ と記す．$\alpha \in \mathcal{R}$ と $d \in \mathcal{D}$ との合成写像を $d \cdot \alpha$ $(\in \mathcal{D})$ と記す．$d \in \mathcal{D}$ と $\psi \in \mathcal{P}$ との合成写像を $\psi * d$ $(\in \mathcal{R})$ と記す．$\psi \in \mathcal{P}$ と $\alpha \in \mathcal{R}$ との合成写像を $\alpha \cdot \psi$ $(\in \mathcal{P})$ と記す．$\psi * d$ を ψ と d との内積と称する．その結果，$\mathcal{D}$ は右 $\mathcal{R}$ 加群，$\mathcal{P}$ は左 $\mathcal{R}$ 加群であり，$\mathcal{P}$ は $\mathcal{D}$ の双対加群となる．

$$\mathcal{R} \cdot \mathcal{P} = \mathcal{P}, \qquad \mathcal{P} * \mathcal{D} = \mathcal{R}, \qquad \mathcal{D} \cdot \mathcal{R} = \mathcal{D},$$
$$\mathcal{R} \cdot \mathcal{R} = \mathcal{R}, \qquad \mathcal{P} = \mathrm{Hom}_{\mathcal{R}}(\mathcal{D}, \mathcal{R})$$

$\mathcal{D}$ から $\mathcal{D}$ への左 $\mathcal{R}$ 準同形写像の全体を $\mathcal{M}$ と記す．$\mathcal{M}$ は環となる．これは $\mathcal{P}$ から $\mathcal{P}$ への右 $\mathcal{R}$ 準同形写像の全体と同一視できる．

$$\mathcal{M} = \mathrm{End}_{\mathcal{R}}(\mathcal{D}) = \mathrm{End}_{\mathcal{R}}^{Q}(\mathcal{P})$$

$m \in \mathcal{M}$ による $d \in \mathcal{D}$ の像を $m * d$ と，$\psi \in \mathcal{P}$ の像を $\psi * m$ と記す．$\mathcal{M}$ の乗法を "$*$" で表す．$\mathcal{D}$ から $\mathcal{R}$ への $\mathcal{R}$ 準同型写像 $\psi \in \mathcal{P}$ と $\mathcal{R}$ から $\mathcal{D}$ への準同型写像 $d \in \mathcal{D}$ との合成写像を $d \cdot \psi$ $(\in \mathcal{M})$ と記す．

◆左単一化関数　左単一化関数とは，R_T から R_T への準同型写像 $(\in \mathcal{R})$

$$l(\hat{s};\hat{t}) : \beta \mapsto l(\hat{s};\hat{t}) \cdot \beta \quad (\beta \in \mathsf{R}_T)$$

であって，次によって定義される．($\tau, \rho \in T$, $\beta_1, \beta_2 \in \mathsf{R}_T$)

$$\begin{cases} l(\hat{s};\hat{t}) \cdot \tau \\ \quad = \begin{cases} \rho \cdots \mathbf{if}\ \exists \rho \in T[\ (\hat{s}\rho = \hat{t}\tau) \wedge (\boldsymbol{X}(\rho) = \boldsymbol{Y}(\hat{s}))\ ] \\ 0 \cdots \mathbf{if}\ \forall \rho \in T[\ \hat{s}\rho \neq \hat{t}\tau\ ] \end{cases} \\ l(\hat{s};\hat{t}) \cdot (\beta_1 + \beta_2) = l(\hat{s};\hat{t}) \cdot \beta_1 + l(\hat{s};\hat{t}) \cdot \beta_2. \end{cases}$$

左単一化関数の全体を L と記す．$\mathcal{R}$ の単位元 1 を L に含める．

$$L = \left\{ l(\hat{s};\hat{t}) \ \middle|\ \hat{s},\ \hat{t} \in H^n\ (n \geq 0) \right\} \cup \{1\}$$

$l(\hat{u};\hat{v})$ と $l(\hat{s};\hat{t})$ の $\mathcal{R}$ での積は次の合成写像である．

$$\left(l(\hat{u};\hat{v}) \cdot l(\hat{s};\hat{t}) \right) \cdot \tau = l(\hat{u};\hat{v}) \cdot \left(l(\hat{s};\hat{t}) \cdot \tau \right)$$

L はこの合成に関して閉じていない．ところが，任意の左単一化関数，および任意個の左単一化関数の合成写像は次の積標準形に変換できる．

$$l(\hat{X};\hat{t}) \cdot l(\hat{u};\hat{Y})$$

これにより $L \cdot L$ は写像の合成に関して閉じる．

次によって表現 E への左単一化関数の作用を定める．

$$\forall \tau \in T[\ (E \cdot l(\hat{s};\hat{t}))\tau = E(l(\hat{s};\hat{t}) \cdot \tau)\ ]$$

すると一般に次の総代入解釈が成り立つ．

$$P(\hat{X}) \cdot l(\hat{X};\hat{t}) = P(\hat{t})$$

◆環 R　$L \in \mathcal{R}$ から生成する $\mathcal{R}$ の部分環を R と定義する．

$$\mathsf{R} \stackrel{\mathrm{def}}{=} \left\{ \sum_i x_i \cdot r_i \ \middle|\ x_i \in L \cdot L,\ r_i \in \boldsymbol{Q} \right\}$$

次は R の元の例である．

$$\alpha_1 = l(f(X);Y)$$
$$\alpha_2 = l(X;Y) + l(X;a) \cdot l(a;Y)$$
$$\alpha_3 = l(X,Y;X,Y) - l(X,Y;U,U) \cdot l(U,U;X,Y)$$
$$\alpha_4 = \frac{7}{13} + l(X;a) \cdot \frac{2}{3} - l(Y;f(b))$$

◆推論加群 D と事実加群 Ψ　素論理式 $P_k(\hat{X}_k) \in G$ $(k \in K_0)$ に基礎総代入 $\tau \in T$ が作用すれば基礎式 $P_k(\hat{X}_k) \cdot \tau \in G_T \subset \mathsf{D}_T$ を得る．これを，$P_k(\hat{X}_k)$ は T から D_T への写像である，と見ることができる．さらに $P_k(\hat{X}_k)$ を R_T から D_T への左 $\boldsymbol{Q}$ 準同形写像へと延長することができる．これにより $P_k(\hat{X}_k)$ は $\mathcal{D}$ の元となる．そこで $\mathcal{R}$ 加群 $\mathcal{D}$ において，$P_k(\hat{X}_k)$ $(k \in K_0)$ を生成元として，$\mathsf{R}(\subset \mathcal{R})$ によって生成する有限生成右 R 加群を，**推論加群 D** と定義する．

$$\mathsf{D} = \left\{ \sum_{k \in K_0} P_k(\hat{X}_k) \cdot \alpha_k \ \middle|\ \alpha_k \in \mathsf{R} \right\} \subset \mathcal{D}$$

D の元を文と称する．例えば次は D の元である．

$$d_1 = P(X)$$
$$d_2 = Q(f(X,a),Y) = Q(X,Y) \cdot l(X,Y;f(X,a),Y)$$
$$d_3 = P(g(Y)) \cdot l(f(X);Y) \cdot \frac{5}{3} - Q(f(Y),a)$$

また，素事実 $\tilde{P}_k(\hat{t}) \in \tilde{G}$ に，次により $\mathsf{D} \subset \mathcal{D}$ から $\mathsf{R} \subset \mathcal{R}$ への R 準同型写像 (内積) 機能を与える．

$$\begin{cases} \tilde{P}_k(\hat{t}) * (P_j(\hat{s}) \cdot \alpha) \stackrel{\mathrm{def}}{=} l(\hat{t};\hat{s}) \cdot \alpha \cdot \delta_{kj}, \\ \tilde{P}_k(\hat{t}) * (d_1 + d_2) \stackrel{\mathrm{def}}{=} \tilde{P}_k(\hat{t}) * d_1 + \tilde{P}_k(\hat{t}) * d_2. \end{cases}$$

これにより $\tilde{P}_k(\hat{t})$ は $\mathcal{P}$ の元となる．そこで $\tilde{G}$ から生成される $\mathcal{P}$ の部分左 R 加群を**事実加群 Ψ** と称する．次は Ψ の元の例である．

$$\psi_1 = \tilde{P}(X)$$
$$\psi_2 = \tilde{Q}(f(X,a),Y)$$
$$\psi_3 = \frac{5}{3} \cdot l(X;f(Y)) \cdot \tilde{P}(g(Y)) - \tilde{Q}(f(Y),a)$$

上記の内積機能の定義から，次の逆代入解釈が成り立つ．

$$l(\hat{u};\hat{Y})\cdot\tilde{P}(\hat{Y})=\tilde{P}(\hat{u})$$

そこで，Ψ は $\tilde{P}_k(\hat{X}_k)$ $(k\in K_0)$ を生成元として $\mathrm{R}(\subset\mathcal{R})$ によって生成される有限生成左 R 加群となる．

$$\Psi=\left\{\sum_{k\in K_0}\alpha_k\cdot\tilde{P}_k(\hat{X}_k)\ \middle|\ \alpha_k\in\mathrm{R}\right\}$$

◆環 M　$d\cdot\psi\in\mathcal{M}$ $(d\in\mathrm{D},\psi\in\Psi)$ の全体から生成する $\mathcal{M}$ の部分環を M と記す．

$$M\overset{\mathrm{def}}{=}\left\{\sum_i d_i\cdot\psi_i\ \middle|\ d_i\in\mathrm{D},\ \psi_i\in\Psi\right\}$$

M の単位元 1 は具体的に次のように書ける．

$$1=\sum_{k\in K_0}P_k(\hat{X}_k)\cdot\tilde{P}_k(\hat{X}_k)$$

◆無限和の収束　推論加群系における無限和の「収束」を定義する．

R の元の無限和

$$\sum_{i=0}^{\infty}\alpha_i$$

は，任意の $\tau\in T$ に対応してある n が存在し，$i>n$ ならば $\alpha_i\cdot\tau=0$ であるとき，収束すると言う．例えば，

$$\sum_{i=0}^{\infty}l(f(X);X)^i=1+l(f(X);X)+l(f(f(X));X)+\cdots$$

は収束するが，次は収束しない．

$$\sum_{i=0}^{\infty}l(X;f(X))^i=1+l(X;f(X))+l(X;f(f(X)))+\cdots$$

D の元の無限和

$$\sum_{i=0}^{\infty}d_i$$

は，任意の $\tau\in T$ に対応してある n が存在して，$i>n$ ならば $d_i\cdot\tau=0$ であるとき，収束すると言う．収束する無限和の基礎例は有限和で表現される．

Ψ の元の無限和

$$\sum_{i=0}^{\infty}\psi_k$$

は，任意の基礎文 $d\in\mathrm{D}$ に対応してある n が存在して，$i>n$ ならば $\psi_i*d=0$ であるとき，収束すると言う．

◆無限べき級数の表記法　$\alpha\in\mathrm{R}$ の無限べき級数を

$$\sum_{i=1}^{\infty}\alpha^i\quad を\quad \{\alpha\}$$

と記す．以後この形の無限べき級数を単に無限級数と言う．

3　R における演算規則

◆左単一化関数の直感的解釈，項加工モデル　総代入 $\theta=\langle f(Y)/X\rangle$ を考えると，これは表現 E の右から作用して E 中の変数記号 X を項 $f(Y)$ で置き換える機能を持つ．ここでさらに右から別の総代入が作用すれば，この変数記号 Y は別の項に置き換えられる．いま変数記号 Y が将来置き換えられる項を s とすれば，総代入 θ は右から Y として渡される項 s に関数記号 f を『着せて』，それを X として右に渡す機能を持つと解釈できる．このように総代入を，右から渡された項になにがしかの変形を加えて左に渡すものと見たとき，この変形操作として総代入に許されているのは，関数記号を『着せる』機能だけであって，関数記号を『脱がす』機能は持っていない．

これに対して左単一化関数 $l(f(X);Y)$ は，右から Y として渡された項 s が関数記号 f を着ていたら f を『脱がし』て X として左に渡す機能を持つと解釈できる．このときもし s が f を着ていなかったら，0 となる．表現への右作用として $l(X;f(Y))$ は総代入 $\langle f(Y)/X\rangle$ と等価であるから，左単一化関数は『変数記号を変更する機能』，『関数記号を着せる機能』，『関数記号を脱がす機能』を有するものと解釈できる．

このように左単一化関数とそれから生成する R の元に対して項を加工する機能で捉える見方を，項加工モデルと言うことにする．この捉え方によれば，本節で述べる R での演算規則も直感的に理解できよう．また，次節以降で展開する R の元自体の計算モデルとしての能力は，この項加工モデルの直接の利用である．

◆左単一化関数の間の関係　左単一化関数の定義から，次の関係が成り立つ．ただし，f を任意の関数記号とする．また $\hat{s}_1$ と $\hat{t}_1$，$\hat{s}_2$ と $\hat{t}_2$，$\hat{X}$ と $\hat{Y}$，はそれぞれ同じ長さの項列とし，空列の場合（長さが 0）を含むものとする．

（項列内の順序の変更）

$$l(\hat{s}_1,\hat{s}_2;\hat{t}_1,\hat{t}_2)=l(\hat{s}_2,\hat{s}_1;\hat{t}_2,\hat{t}_1).$$

（項列内の重複の除去）

$$l(\hat{s}_1,\hat{s}_1,\hat{s}_2;\hat{t}_1,\hat{t}_1,\hat{t}_2)=l(\hat{s}_1,\hat{s}_2;\hat{t}_1,\hat{t}_2).$$

（共通記号の除去）　$u\in H_T$，$t[\hat{X}]\in H$ とすると，

$$\begin{aligned}l(u,\hat{s}_1;u,\hat{t}_1)&=l(\hat{s}_1;\hat{t}_1),\\ l(f(\hat{s}_1),\hat{s}_2;f(\hat{t}_1),\hat{t}_2)&=l(\hat{s}_1,\hat{s}_2;\hat{t}_1,\hat{t}_2),\\ l(t[\hat{X}],\hat{s}_1;t[\hat{Y}],\hat{t}_1)&=l(\hat{X},\hat{s}_1;\hat{Y},\hat{t}_1).\end{aligned}$$

（無効果要素の除去）　$\theta=\{t_0/X\}$ と置くと，

$$l(X,\hat{s}_1;t_0,\hat{t}_1)=l(\hat{s}_1\theta\,;\hat{t}_1).$$

◆左単一化関数の積の変形規則　左単一化関数の積において次の変形規則が成り立つ．

(中間変数記号の同時変更)　$\hat{X}$ と $\hat{Y}$ はそれぞれ重複のない同長の変数記号列で，$\{\hat{Y}\} \cap \boldsymbol{Y}(\hat{t}, \hat{u}) = \emptyset$ ならば，

$$l(\hat{s};\hat{t}) \cdot l(\hat{u};\hat{v}) = l\Big(\hat{s};\hat{t}\Big\{\frac{\hat{Y}}{\hat{X}}\Big\}\Big) \cdot l\Big(\hat{u}\Big\{\frac{\hat{Y}}{\hat{X}}\Big\};\hat{v}\Big).$$

(総代入解釈)　$\hat{X}$ に変数記号の重複がなければ，

$$l(\hat{s};\hat{t}) \cdot l(\hat{X};\hat{v}) = l\Big(\hat{s};\hat{t}\Big\langle\frac{\hat{v}}{\hat{X}}\Big\rangle\Big).$$

(逆代入解釈)　$\hat{X}$ に変数記号の重複がなく，$\hat{Z}$ が含む変数記号を $\hat{u}$ 中の変数記号とは異なるように変更したものを $\hat{Z}'$ とすると，

$$l(\hat{s}[\hat{Z}];\hat{X}) \cdot l(\hat{u};\hat{v}) = l(\hat{Z};\hat{Z}') \cdot l\Big(\hat{u}\Big\{\frac{\hat{s}[\hat{Z}']}{\hat{X}}\Big\};\hat{v}\Big).$$

特に $\boldsymbol{Y}(\hat{u}) = \{\hat{X}\}$ ならば

$$l(\hat{s};\hat{X}) \cdot l(\hat{u};\hat{v}) = l\Big(\hat{u}\Big\{\frac{\hat{s}}{\hat{X}}\Big\};\hat{v}\Big).$$

(無効果要素の除去)　$X \notin \boldsymbol{Y}(\hat{t}, \hat{u})$ ならば，

$$l(\hat{s};\hat{t}) \cdot l(X,\hat{u};t,\hat{v}) = l(\hat{s};\hat{t}) \cdot l(\hat{u};\hat{v}),$$
$$l(t_0,\hat{s};X,\hat{t}) \cdot l(\hat{u};\hat{v}) = l(\hat{s};\hat{t}) \cdot l(\hat{u};\hat{v}).$$

$Z \notin \boldsymbol{Y}(\hat{s}) \wedge X \notin \boldsymbol{Y}(\hat{t}, \hat{u})$ ならば，

$$l(Z,\hat{s};X,\hat{t}) \cdot l(\hat{u};\hat{v}) = l(\hat{s};\hat{t}) \cdot l(\hat{u};\hat{v}).$$

特に $\{\hat{X}\} \cap \{\hat{Y}\} = \{\hat{U}\}$ ならば，

$$l(\hat{X};\hat{X}) \cdot l(\hat{Y};\hat{Y}) = l(\hat{U};\hat{U}).$$

(共通変数の統合)

$$l(\hat{t};\hat{u}) \cdot l(p,q,\hat{v};X,X,\hat{w}) = \begin{cases} 0 & \cdots\cdots\cdots\cdots \textbf{if } p\text{と}q\text{は単一化不能}, \\ l(\hat{t};\hat{u}\sigma) \cdot l(r,\hat{v}\sigma;X,\hat{w}) & \cdots \textbf{if } p\text{と}q\text{は単一化可能}. \end{cases}$$

（ただし σ は p と q との最汎単一化作用素 $(\in\Theta)$，r は $p\sigma(=q\sigma)$ とする）

(項の移動)　$s[\hat{Z}]$ が含む変数記号を $\hat{v}$ および $\hat{u}$ 中の変数記号と重複がないように変更したものを $s[\hat{Z}']$，$\sigma = \{s[\hat{Z}']/Y\}$ とすれば，

$$l(\hat{t};\hat{u}) \cdot l(Y,\hat{v};s[\hat{Z}],\hat{w}) = l(\hat{t};\hat{u}\sigma) \cdot l(\hat{Z}',\hat{v}\sigma;\hat{Z},\hat{w}).$$

◆無限級数との乗算　無限級数 $\{\alpha\}$ が収束するとき，次が成り立つ．

$$\{\alpha\} \cdot \alpha = \{\alpha\} - \alpha, \qquad \alpha \cdot \{\alpha\} = \{\alpha\} - \alpha$$

$$\{\alpha\} \cdot \beta = \alpha \cdot \beta + \{\alpha\} \cdot (\alpha \cdot \beta)$$

4　R の元による手続の模倣

前節では環 R の演算規則を議論した．ここでは R の元に対する項加工モデルの応用として，任意の手続を環 R の元で模倣する可能性に関して考察する．

◆通過項の処置　R の元の基本要素である左単一化関数の働きは，項加工モデルで捉えることができる．この時，左単一化関数の右から入力される総代入の指定変数の中に，その左単一化関数の代表変数に含まれないものがあると，その指定変数とともに運ばれてくる項は棄却されてしまう．例えば

$$l(f(X);X) \cdot \Big\langle \frac{f(f(a))}{X}, \frac{g(b)}{I} \Big\rangle = \Big\langle \frac{f(a)}{X} \Big\rangle$$

のように，$g(b)$ の情報が落ちてしまう．そこで，その元の主たる加工内容にかかわらない，ただ通過すればよい項 (通過項) を通過させるために，通過変数を用意する．例えば上例では $l(f(X);X)$ の代わりに通過変数 I を用いて $l(I,f(X);I,X)$ とする．

$$l(I,f(X);I,X) \cdot \Big\langle \frac{f(f(a))}{X}, \frac{g(b)}{I} \Big\rangle = \Big\langle \frac{f(a)}{X}, \frac{g(b)}{I} \Big\rangle$$

いま，β は X に目的の加工をし，通過変数 I を持つとする．この β を利用し，項 Z に目的の加工をし，項 U, V は素通りさせたい場合は，次のようにする．

$$l(z(U,V),X;I,X) \cdot \beta \cdot l(I,X;z(U,V),X)$$

このとき，$l(I,X;z(U,V),X)$ を通過項の「統合処理」，$l(z(U,V),X;I,X)$ を通過項の「復元処理」と称する．この一般的処方箋により，通過変数は 1 個あれば十分である．

◆計算モデルとしての R の元の設計規約　以上の対処法に基づいて，以下では計算モデルとしての R の元は，次の規約を守って構成するものとする．

(1)　通過変数は I であるとする．

(2)　統合処理のための関数記号としては z を用いる．

(3)　引数 (入力) は変数記号 X で受け取るものとする．また引数が複数個の場合は関数記号 s を用いて，項 $s(X_1,\cdots,X_n)$ の形で受け取るものとする．

(4)　計算結果は常に変数記号 X で引き渡すものとする．

◆再帰的な定義と無限級数　再帰的に定義される関数を計算する R の元は，それ自身

$$\alpha = \gamma + \beta \cdot \alpha \cdot \delta$$

のような再帰的関係を満たす．例えば計算の対象となる項が，対象記号として a と b だけを，関数記号としては arity= 1の f だけを含むとき，与えられた項中の a を b で置き換える機能を持つRの元 α は，次の β,γ,δ を用いて，上記のように再帰的に定義できる．

$$\begin{cases}\beta = l(I,X;I,f(X)),\\ \gamma = l(I,X;I,b)\cdot l(I,a;I,X),\\ \delta = l(I,f(X);I,X).\end{cases}$$

この例のように，t を基礎項とすると，$\gamma\cdot l(I,X;I,t)$ と $\delta\cdot l(I,X;I,t)$ とは，一方が非0であれば他方は0であるということが成り立つ場合が多い．つまり両方とも非0ということはない．これを γ と δ の排他条件という．

この再帰的定義にしたがって，実際に与えられた項の計算を行うことができる．例えば，項 $t=f(f(a))$ に対して

$$\alpha\cdot l(I,X;I,t)$$

の計算は，次のように次々と α に再帰的定義を代入することで実行できる．

$$\begin{aligned}&\alpha\cdot l(I,X;I,f(f(a)))\\ &=\underbrace{\gamma\cdot l(I,X;I,f(f(a)))}_{=0}+\beta\cdot\alpha\cdot\underbrace{\delta\cdot l(I,X;I,f(f(a)))}_{=l(I,X;I,f(a))}\\ &=\beta\cdot\underbrace{\alpha}_{\text{代入}}\cdot l(I,X;I,f(a))\\ &=\beta\cdot\underbrace{\gamma\cdot l(I,X;I,f(a))}_{=0}+\beta^2\cdot\alpha\cdot\underbrace{\delta\cdot l(I,X;I,f(a))}_{=l(I,X;I,a)}\\ &=\beta^2\cdot\underbrace{\alpha}_{\text{代入}}\cdot l(I,X;I,a)\\ &=\beta^2\cdot\underbrace{\gamma\cdot l(I,X;I,a)}_{=l(I,X;I,b)}+\beta^3\cdot\alpha\cdot\underbrace{\delta\cdot l(I,X;I,a)}_{=0}\\ &=\beta^2\cdot l(I,X;I,b)\\ &=l(I,X;I,f(f(b))).\end{aligned}$$

このような計算を，再帰的定義に忠実な計算と言うことにする．このような計算を行う元 α は，Rの中では次のように無限和の形で表わされる．

$$\begin{aligned}\alpha&=\gamma+\beta\cdot\alpha\cdot\delta\\ &=\gamma+(\beta)\cdot\gamma\cdot(\delta)+(\beta)^2\cdot\alpha\cdot(\delta)^2\\ &=\gamma+(\beta)\cdot\gamma\cdot(\delta)+(\beta)^2\cdot\gamma\cdot(\delta)^2+(\beta)^3\cdot\alpha\cdot(\delta)^3\\ &=\cdots\\ &=\sum_{i=0}^{\infty}\beta^i\cdot\gamma\cdot\delta^i\qquad(=\gamma+\sum_{i=1}^{\infty}\beta^i\cdot\gamma\cdot\delta^i)\end{aligned}$$

上記の α の再帰的定義に忠実な計算の停止性と，対応する無限和の収束性に関して，次の補題が成り立つ．

◇補題 1　関数 $\alpha\in$ R の再帰的定義

$$\alpha=\gamma+\beta\cdot\alpha\cdot\delta \tag{1}$$

に忠実な計算が停止性を持つならば，無限和

$$\sum_{i=0}^{\infty}\beta^i\cdot\gamma\cdot\delta^i \tag{2}$$

は収束する．

［証明］　t を任意の基礎項とすると，計算 $\alpha\cdot l(I,X;I,t)$ が有限ステップで終了するためには，ある n において，$\delta^n\cdot l(I,X;I,t)$ が0とならなければならないことは容易に示せる．これは (2) 式が収束することを意味する．■

◇補題 2　無限和

$$\alpha=\{\delta\}=\sum_{i=1}^{\infty}\delta^i \tag{3}$$

が収束することと，この関数の再帰的関係

$$\alpha=\delta+\alpha\cdot\delta \tag{4}$$

に忠実な計算が停止性を持つこととは同等である．

［証明］　(4) に忠実な計算が常に停止すれば (3) が収束することは，補題1から導かれる．逆に (3) が収束するとする．任意の基礎項 t に対して，ある n が存在して，$i\geq n$ では $\delta^i\cdot l(I,X;I,t)$ は0である．これは (4) に従った $\alpha\cdot l(I,X;I,t)$ の計算における α の代入は，少なくとも n 回目に終了することを意味する．従って (4) に忠実な計算は停止する．■

◆次数対応法　前記の無限和

$$\gamma+\sum_{i=1}^{\infty}\beta^i\cdot\gamma\cdot\delta^i$$

は，そのままでは前節で導入した無限級数の記法

$$\{\alpha\}=\sum_{i=1}^{\infty}\alpha^i$$

で表わすことはできない．そこでこれを2個の無限級数の積で表わすことを考える．

もし，上記の例のように β と γ と δ が通過変数 I をそのまま通過させるならば，上記の α は無限級数の記法を用いて，次のように表現することができる．

$$\alpha=\gamma+\kappa_*\cdot\{\beta\cdot\eta_*\}\cdot\gamma\cdot\{\eta\cdot\delta\}\cdot\kappa$$

ただし，g を arity= 1の関数記号とすると，

$$\begin{cases}\kappa_*=l(z(I,a),X;I,X),\\ \kappa=l(I,X;z(I,a),X),\\ \eta_*=l(I,X;z(I,U),X)\cdot l(z(I,g(U)),X),\\ \eta=l(I,X;z(I,g(U)),X)\cdot l(z(I,U),X;I,X).\end{cases}$$

これが成り立つ理由は以下の通り．β と γ と δ は，変数 X について加工し，変数 I を素通りさせる．一方 κ_* と η_* と η と κ は，変数 X を素通りさせ変数 I を加工する．

よってこれら第1のグループの元と第2のグループの元とは可換である．ゆえに次が成り立つ．

$$(\beta\cdot\kappa_*)^n=\beta^n\cdot{\kappa_*}^n,\quad(\eta\cdot\delta)^m=\eta^m\cdot\delta^m,$$
$$\delta^m\cdot\kappa=\kappa\cdot\delta^m,\qquad\kappa_*\cdot\beta^n=\beta^n\cdot\kappa_*,$$
$$\gamma\cdot\eta^m=\eta^m\cdot\gamma,\qquad\gamma\cdot\kappa=\kappa\cdot\gamma.$$

したがって

$$\begin{aligned}\kappa_*\cdot(\beta\cdot\eta_*)^n\cdot\gamma\cdot(\eta\cdot\delta)^m\cdot\kappa&\\=\kappa_*\cdot\beta^n\cdot{\eta_*}^n\cdot\gamma\cdot\eta^m\cdot\delta^m\cdot\kappa&\\=\beta^n\cdot\kappa_*\cdot{\eta_*}^n\cdot\eta^m\cdot\kappa\cdot\gamma\cdot\delta^m&.\end{aligned}$$

さらに，

$$\eta_*\cdot\eta=l(I,X;I,X)\quad\text{および}\quad\kappa_*\cdot\kappa=l(I,X;I,X)$$

が成り立つから，

$$\kappa_*\cdot{\eta_*}^n\cdot\eta^n\cdot\kappa=l(I,X;I,X)$$

である．また

$$n\geq1\quad\Rightarrow\quad0=\kappa_*\cdot{\eta_*}^n\cdot\kappa=\kappa_*\cdot\eta^n\cdot\kappa$$

が成り立つように η_* と η とが作られているから

$$n\neq m\quad\Rightarrow\quad\kappa_*\cdot{\eta_*}^n\cdot\eta^m\cdot\kappa=0$$

が成り立つ．以上より

$$\kappa_*\cdot(\beta\cdot\eta_*)^n\cdot\gamma\cdot(\eta\cdot\delta)^m\cdot\kappa=\begin{cases}0&\cdots\text{if }n\neq m,\\\beta^n\cdot\gamma\cdot\delta^n&\cdots\text{if }n=m.\end{cases}$$

が成り立つから，

$$\kappa_*\cdot\left(\sum_{n=1}^{\infty}(\beta\cdot\eta_*)^n\right)\cdot\gamma\cdot\left(\sum_{m=1}^{\infty}(\eta\cdot\delta)^m\right)\cdot\kappa=\sum_{n=1}^{\infty}\beta^n\cdot\gamma\cdot\delta^n$$

である．

もし β か δ が通過変数 I を利用している場合には，変数 U を新たな通過変数として位置付け，これを素通りするように β と δ を変形し，κ_* と η_* と η と κ は，次のように，変数 U を利用し，変数 I は素通りさせるように構成し，

$$\begin{cases}\kappa_*=l(a,I,X;U,I,X),\\\kappa=l(U,I,X;a,I,X),\\\eta_*=l(g(U),I,X;U,I,X),\\\eta=l(U,I,X;g(U),I,X),\end{cases}$$

さらに次のような ξ_* と ξ を用いればよい．

$$\begin{cases}\xi_*=l(z(I,U),X;I,X),\\\xi=l(I,X;z(I,U),X).\end{cases}$$

すると α は次のように表わせる．

$$\alpha=\gamma+\kappa_*\cdot\{\beta\cdot\eta_*\}\cdot\xi_*\cdot\gamma\cdot\xi\cdot\{\eta\cdot\delta\}\cdot\kappa$$

この場合は，前記の議論において γ を $\xi_*\cdot\gamma\cdot\xi$ で置き換えた議論が成り立つことから，この α が正しいことが分かる．

以上の対処法を次数対応法と呼ぶ．

5　Rの元によるチューリングマシンの模倣

チューリングマシンをRの元で模倣する方法を検討する．簡単のために，1テープ，4項系列の遷移規則を持つ決定性チューリングマシンを対象とする．

そのようなチューリングマシンは，テープに書き込める記号の集合 Γ と，マシンの状態の集合 S と，遷移規則

$$\sigma_i:(s_i,\gamma_i,s_i',\gamma_i')\quad(s_i,s_i'\in S,\gamma_i\in\Gamma,\gamma_i'\in\Gamma\cup\{\text{左},\text{右}\})$$

の集合 $\Sigma=\{\sigma_i\}_{i\in I}$ と初期状態 s_0 ($\in S$) と終了状態の集合 Q ($=\{q_j\}_{j\in J}\subset S$) と，によって指定される．テープ上のなにも書かれていない領域には空記号 $*$ ($\in\Gamma$) が書かれているものとする．

Rに於いては Γ と S の要素には，Rで利用可能な基礎項を1対1に対応させる．

$$\mu:\Gamma\to H_T,\qquad\mu:S\to H_T$$

ただし

$$a\neq b\quad\Rightarrow\quad\mu(a)\neq\mu(b)\qquad(a,b\in\Gamma\cup S)$$

とする．このような対応付けは H_T が無限であれば常に可能である．なお以下の記述では，模倣側のRの元などの表現に Γ や S の元 a が直接書かれているときは，実際には $\mu(a)$ の事であるとする，

次に，マシン全体の状態，すなわち，テープの状態とマシンの状態，ならびに読み書きヘッドの位置，の表現には，arity$=2$ の関数記号 g と arity$=4$ の関数記号 h を想定し[1]，g を用いた2個のリスト：

$$A=g(g(g(\cdots g(\text{nil},\gamma_m^L),\cdots),\gamma_2^L),\gamma_1^L),$$
$$B=g(\gamma_1^R,g(\gamma_2^R,\cdots,g(\gamma_n^R,\text{nil})\cdots))$$

を用いて項

$$h(A,s,\gamma,B)$$

で表現する．ただし nil は $\mu(\Gamma)$ と $\mu(S)$ には含まれない基礎項とする．これは，テープ上の記号が

$$\cdots,*,*,\gamma_m^L,\cdots,\gamma_2^L,\gamma_1^L,\gamma,\gamma_1^R,\gamma_2^R,\cdots,\gamma_n^R,*,*,\cdots$$

であって，ヘッドは γ を見ており，マシンの状態は s であることを表わすものとする．A における項 nil はテープ上の有効な記号列の左端を表わす．つまり，これより左は全てブランク $*$ であることを意味する．同様に B における項 nil はテープ上の有効な記号列の右端を表わす．つまり，これより右は全てブランク $*$ であることを意味する．

遷移規則

$$\sigma_i:(s_i,\gamma_i,s_i',\gamma_i')$$

すなわち，マシンの状態が s_i，ヘッド上の記号が γ_i であるならば，テープ上の現在見ている桝目に記号 γ_i' を書き

[1] arity$=4$ の関数記号が存在しなければ，$h(X,Y,Z,W)$ の代わりに $g(g(X,Y),g(Z,W))$ を用いてもよい．

込んで，状態を s'_i に変えるという遷移規則は，次のようなRの元で表わされる．

$$\beta_i = l(I,X;I,h(L,s'_i,\gamma'_i,R)) \cdot l(I,h(L,s_i,\gamma_i,R);I,X).$$

マシンの全状態が t で表わされるとき，$\beta_i \cdot l(I,X;I,t)$ を計算すると，この遷移規則 σ_i の条件に t が適合していなければ0，適合していれば $l(I,X;I,t')$ なる左単一化関数となり，この時 t' は σ_i による遷移結果に対応する．

同様に，遷移規則

$$\sigma_i : (s_i, \gamma_i, s'_i, \text{右})$$

すなわち，マシンの状態が s_i，ヘッド上の記号が γ_i であるならば，ヘッドを右に移動して状態を s'_i に変えるという遷移規則は次のようなRの元で表わされる．

$$\begin{aligned}\beta_i = {} & l(I,X;I,h(g(L,\gamma_i),s'_i,U,R)) \\ & \cdot l(I,h(L,s_i,\gamma_i,g(U,R));I,X) \\ & + l(I,X;I,h(g(L,\gamma_i),s'_i,*,\mathrm{nil})) \\ & \cdot l(I,h(L,s_i,\gamma_i,\mathrm{nil});I,X).\end{aligned}$$

同様に，遷移規則

$$\sigma_i : (s_i, \gamma_i, s'_i, \text{左})$$

すなわち，マシンの状態が s_i，ヘッド上の記号が γ_i であるならば，ヘッドを左に移動して状態を s'_i に変えるという遷移規則は，次のようなRの元で表わされる．

$$\begin{aligned}\beta_i = {} & l(I,X;I,h(L,s'_i,U,g(\gamma_i,R))) \\ & \cdot l(I,h(g(L,U),s_i,\gamma_i,R);I,X) \\ & + l(I,X;I,h(\mathrm{nil},s'_i,*,g(\gamma_i,R))) \\ & \cdot l(I,h(\mathrm{nil},s_i,\gamma_i,R);I,X).\end{aligned}$$

模倣対象は決定性チューリングマシンであるから，どの状態においても，2個以上の遷移条件が適合することはない．従って

$$\beta = \sum_{i \in I} \beta_i$$

と置くと，マシンの全状態が t で表わされるとき，

$$\beta \cdot l(I,X;I,t)$$

を計算すると，どの遷移条件も適合しないとき0，どれかの遷移条件が適合すれば左単一化関数 $l(I,X;I,u)$ を得るが，ここで得られた u はマシンの次の全状態を表わすことになる．従って，初期状態におけるマシンの全状態が t_0 で表わされるとすると，状態遷移を1回，2回，3回，…と繰り返した結果を含む左単一化関数の総和は無限級数を用いて

$$\{\beta\} \cdot l(I,X;I,t_0)$$

によって得られる．

また，終了状態 q_j に達した事を判定する元は

$$\epsilon_j = l(I,X;I,h(L,q_j,U,R)) \cdot l(I,h(L,q_j,U,R);I,X)$$

によって与えられる．すなわち，マシンの全状態が t で表わされるとき，

$$\epsilon_j \cdot l(I,X;I,t)$$

を計算すると，終了状態 q_j に達しているとき $l(I,X;I,t)$，達していないとき0となる．従っていずれかの終了状態に達していることを判定する元は

$$\epsilon = \sum_{j \in J} l(I,X;I,h(L,q_j,U,R)) \cdot l(I,h(L,q_j,U,R);I,X)$$

で表わされる．すなわち

$$\epsilon \cdot l(I,X;I,t)$$

は，t がいずれかの終了状態であるとき $l(I,X;I,t)$，そうではないときは0となる．そこで，和 $\{\beta\} \cdot l(I,X;I,t_0)$ の中で，終了状態に達したものを含む左単一化関数は，この ϵ を乗ずることで抽出できる．従って，チューリングマシン全体は次の無限級数で表現できる．

$$\alpha = \epsilon \cdot \{\beta\}$$

この無限級数は多重の無限級数を含んでいない．

このように，Rの元によって任意の1テープの決定性チューリングマシンを模倣できる．これはRが計算モデルとして十分強力であり，計算可能な全ての手続をその元として含んでいることを意味する．また，推論加群系における無限級数の収束性問題は，チューリングマシンの停止性問題に還元されるので，決定不能であることが分かる．さらに，推論加群系における2個の要素の同定問題は，2個のチューリングマシンの同定問題に帰着されるので，これも決定不能であることが分かる．

6 考察

環Rの元はTMを模倣できるから，一般帰納的関数や項書換系(TRS)をも模倣できる．この模倣を環Rの元で直接行うことを試みた．その結果，TMが1重の無限級数で模倣されるのに対して，TRSの直接模倣では

$$\{\alpha + \{\beta\}\}$$

のような無限級数の無限級数が出現し，一般帰納的関数の直接模倣では原始帰納法と一般帰納法の利用の回数だけの入れ子になった無限級数が出現する．しかし，これらがTMで模倣できることから，Rの無限級数の入れ子を1重の無限級数に帰着させることができるはずである．実際，これは可能であることを示すことができる．例えば一般に

$$\alpha = \{\beta + \epsilon \cdot \{\gamma\}\} \cdots\cdots \text{右標準表現}$$

の場合は，

$$\alpha = \zeta + (\zeta \cdot \{\zeta + \gamma\}), \qquad \zeta = \beta + \epsilon \cdot \gamma - \beta \cdot \gamma.$$

と1重化される．またより一般に

$$\alpha = \{\beta + \epsilon \cdot \{\gamma\} \cdot \eta\}$$

を左標準表現に変換する方法も存在する．ただし，$\sum_{i=1}^{\infty} \beta^i$ は収束しないが $\sum_{i=1}^{\infty} \alpha \cdot \beta^i$ は収束する場合に対し

$$(\alpha \cdot \{\beta\}) \stackrel{\text{def}}{=} \sum_{i=1}^{\infty} \alpha \cdot \beta^i$$

なる表記法を追加する必要がある．

本稿では環Rの元の計算能力を取り上げたが，環 M でもまったく同様の事が成りたつ．それは $\alpha \in$ Rがある計算を模倣するならば，$P(\hat{X}) \cdot \alpha \cdot \tilde{P}(\hat{X}) \in M$ も同じ計算を模倣するからである．

環Rおよび M の元がTMを模倣できるということは，推論加群系において，2個の元の同等性を判定する問題が決定不能であることを意味する．これは推論加群系の数式処理系を構成する上での一つの困難を与える．例えば $\alpha - \beta$ が0かどうかを判定するためには α と β の同等性が判定できなければならないからである．無限級数を含まない元ではこれを判定する手順が存在する．しかし，無限級数を含む場合には，本当は0に等しい元を，0ではない表現のまま残して計算を続けることを容認しなければならない．

推論加群系は推論を代数化することを目的に構成されており，この代数系の上で証明問題，充足可能性問題を表現できること，また，節集合によって任意の手続を模倣できることから，本稿で述べたことは自明であると考えるかもしれないが，それはあたらない．実際，[3, 5] に述べた通り，推論を代数化し，充足可能性をその上で表現するだけであれば，左単一化関数ではなく，代入から生成する環 R で十分である．しかし代入およびそれから生成する環 R の元では，項の加工能力としては，関数記号を被せる機能しかないため，十分な計算能力を持たない．左単一化関数の導入はこの代数系を双対な形にするために必要であったが，同時にその係数環Rが本稿で述べたような計算能力を持つ上でも必要なのである．

7 おわりに

推論加群系の環Rの元の計算モデルとしての能力について考察し，環Rの元がチューリングマシンを模倣できることを示した．その結果，無限級数の収束の判定問題は決定不能であることと，2個の元の同等性の判定問題は決定不能であることが得られた．これらは推論加群系のための完全な数式処理系を実現することには限界があることを示している．また，多重の無限級数を1重の無限級数に変換できることを示した．

今後の課題として，(汎) 双対元の構成問題と，実用上の観点からは十分であるような，推論加群系の数式処理系の構成方法，などが残されている．

参考文献

[1] 山崎勇：推論の代数化と代数的証明原理, 第2回人工知能学会全国大会, 1-3, pp.27-30, (1988).

[2] 山崎勇：充足可能性問題の代数化－同次証明方程式による判定法，第40回情報処理全国大会 (I)，7c-7, pp.210–211, (1990).

[3] 山崎勇：一階述語論理における代数的証明原理, 人工知能学会誌, V0l.5, No.3, pp.279 -290, (1990).

[4] 山崎勇：証明方程式の線形時間解法アルゴリズム, 情報処理学会アルゴリズム研究会, AL28-2, pp.9 -16, (1992).

[5] 山崎勇：一階述語論理における推論と充足の代数化, コンピュータソフトウェア, Vol.12, No.2, pp.16 -31, (1995).

[6] 山崎勇：推論加群系と自動証明への応用, (LA シンポジウム'96w)，数理解析研究所講究録950，計算モデルと計算の複雑さに関する研究, pp.174 -179, (1996).

[7] 山崎勇：拡張された推論加群系とその代数，コンピュータソフトウェア，Vol.14，No.2，pp.45 -70, (1997).