

数式処理システム REDUCE を用いた マルコフ連鎖の解析パッケージ MCA

中 村 隆 志

1. ま え が き

確率過程の一つであるマルコフ過程のうち、時点として取り得る値が離散的であり、取り得る状態も離散的なものをマルコフ連鎖と呼ぶ。その性質は「時点 n で状態 i にあるとき、次の時点 $n+1$ で状態 j に推移する確率（推移確率）が時点 $n-1$ 以前の履歴には関係がない。」というものである。マルコフ連鎖はオペレーションズ・リサーチをはじめ、自然科学や社会科学の各分野でさまざまな現象のモデル解析に利用されている⁽¹⁾。

特に、理論的に扱いやすく、応用上もよく用いられているのは、推移確率がいかなる時点においても不変である斉時的マルコフ連鎖である。この場合のマルコフ連鎖の挙動に関する各種特性量の導出に関しては文献(2)、(3)などに詳細に述べられている。これらは大部分が連立方程式や逆行列などの計算によって得られるため、計算機を用いれば容易に数値計算が可能である。しかし、これらの諸量の最適化や感度分析などを行う場合には、推移確率の一部をパラメータとして記号で与え、記号を含んだままの形の解析解を得た方が都合がよい。従来、このような記号を含む計算は手作業で行ってきた。しかし、状態数が増加すると手計算による解の導出は困難となる。

そこで、考えられるのは計算機で数式を記号のまま処理（数式処理）することである。近年、CPUの高速化、メモリ容量の増加、及び記号処理研究の進展に伴い、計算機による数式処理が実用的になってきた。汎用数式処理システムとしてはMACSYMA、REDUCE、muMATHなどがよく知られている^{(4),(5)}。これらを用いれば、特性量を記号を含む形で計算可能である。しかし、数式処

理システム自体がまだ一般になじみが薄く、又、計算のたびにプログラムを作成するのは大変である。

そこで、本研究では数式処理システム REDUCE を用い、推移確率行列に記号を含む場合のマルコフ連鎖の特性量の導出を支援するためのソフトウェア・パッケージ MCA (Markov Chain Analyzer) の開発を行った。

2. MCA の概要

マルコフ連鎖の代表的な型として、吸収的マルコフ連鎖とエルゴード的マルコフ連鎖とがある。他のマルコフ連鎖も推移確率行列の分解などを行うことにより、この二つの型に帰着できる。よく用いられる特性量は次のとおりである。

a) 吸収的マルコフ連鎖

- 状態確率 (時点 n で状態 j にいる確率。この計算には n 次推移確率が必要である。)
- 訪問回数 (吸収されるまでの一時的状態 j の訪問回数) の期待値, 分散
- 吸収時間 (吸収されるまでのステップ数) の期待値, 分散, 及び分布
- 吸収確率 (いつかは吸収状態 j に吸収される確率)

b) エルゴード的マルコフ連鎖

- 状態確率
- 定常分布
- 初到達時間 (first passage time : 状態 j に初めて到達するまでのステッ

表1 MCA

プログラム名	機能
MZTRANS	n 次推移確率の z 変換を数式処理により求める。
AZTRANS	吸収的マルコフ連鎖の吸収時間分布の z 変換を数式処理により求める。
FPZTRANS	エルゴード的マルコフ連鎖の初到達時間分布の z 変換を数式処理により求める。
ABSORB	吸収的マルコフ連鎖の、訪問回数の期待値と分散、吸収時間の期待値と分散、及び吸収確率などを数式処理により求める。
ERGODIC	エルゴード的マルコフ連鎖の、定常分布、初到達時間の期待値と分散などを数式処理により求める。

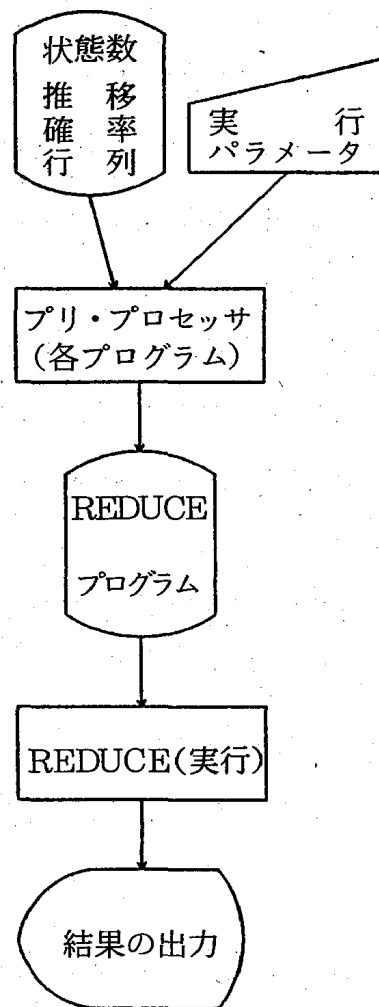


図1 処理形態

ブ数)の期待値, 分散, 及び分布。

本研究で開発したソフトウェア・パッケージ MCA は数式処理により, これらの特性量の導出を支援するためのものであり, 表1に示す五つのプログラムで構成されている。これらは数式処理システムとして, 北海道大学大型計算機センター M-680 H 上で稼動している REDUCE バージョン 3. 2^⑥ を利用している。表1の各プログラムは図1に示すように REDUCE で書かれたプログラムを生成するプリ・プロセッサの形をとる。したがって, 処理は二つのステップに分かれる。まず, 最初にプリ・プロセッサである各プログラムがファイルに用意されている状態数と推移確率行列, 及び実行時にキーボードから入力される実行パラメータなどに従い, 各特性量を求めるために必要な一連の

REDUCE プログラムをファイルに出力する。次に、そのプログラムを REDUCE により実行し、結果を表示する。結果の式は適当な変数に格納されているので、利用者がそれを用いて何らかの処理（たとえば、因数分解、微分、数値計算など）を行うことも可能である。REDUCE には記号行列の計算や連立1次方程式の解法ルーチンが組み込まれているため、各種特性量の計算は比較的容易にできる。生成される REDUCE プログラムは固定的部分が多いが、推移確率行列を与える設定文や、連立方程式の解法のための SOLVE 演算子におけるパラメータの指定の部分などが推移確率行列や実行パラメータにより変化する。

利用者は状態数と推移確率行列を格納したファイル（データセット）を用意し、適当な実行パラメータを入力するだけで結果を得ることができる。推移確率行列の各要素は15文字以内の REDUCE で許される式とする。

なお、各プログラム（プリ・プロセッサ）はすべて VOS 3 PASCAL で記述されている。

3. 各プログラムの仕様

各プログラムの機能、実行パラメータ、計算方法などを示す。

なお、以後の説明では推移確率行列が $P = [p_{ij}]$ で、状態数が N の斉時的な有限マルコフ連鎖を考えている。

3. 1 MZTRANS

(a) 機 能

実行パラメータで指定された初期状態 i に関する n 次推移確率 $p_{ij}(n)$ の z 変換 $P_{ij}(z)$ ($j=1, 2, \dots, N$) を求めて出力する。

(b) 実行パラメータ

初期状態 i を入力する。（リスト形式で複数個の入力が可能である。）

(c) 生成される REDUCE プログラムにおける計算方法

n 次推移確率行列 $P(n)$ の z 変換は

$$P(z) = [P_{ij}(z)] = (I - zP)^{-1} \quad (1)$$

であるから^③、この逆行列を計算すればよいことになる。しかし、REDUCE に組み込まれている自動的な逆行列計算を用いた場合には3メガバイトのメモリを使用しても計算できるのはせいぜい5行×5列程度の行列である。したがって、ここでは次のような方法を採用している。

まず、式(1)に右から $(I-zP)$ を乗ずると

$$[P_{ij}(z)] (I-zP) = I \quad (2)$$

これを行毎に分けると

$$[P_{i1}(z), P_{i2}(z), \dots, P_{iN}(z)] (I-zP) = \eta_i \quad (3)$$

ただし、

η_i : i 番目の要素が1で他はすべて0の行ベクトル

この連立方程式を SOLVE 演算子で計算することにより、指定された初期状態 i に関する $P_{ij}(z)$ ($j=1, 2, \dots, N$) を求めることができる。

3. 2 AZTRANS

(a) 機能

各状態 i から、実行パラメータで指定された吸収状態 a への吸収時間分布の z 変換 $G_{ia}(z)$ 、又は、吸収状態の集合 A への吸収時間分布の z 変換 $G_{iA}(z)$ を求めて出力する。

(b) 実行パラメータ

吸収状態または吸収状態の集合を入力する。吸収状態入力か吸収状態の集合入力かはオプション番号により指定する。

• オプション番号

1 : 吸収状態を入力することを示す。(リスト形式で複数個の吸収状態の入力が可能である。)

2 : 吸収状態の集合を入力することを示す。

(c) 生成される REDUCE プログラムにおける計算方法

i) 吸収状態を入力した場合

状態 i から a への n 次推移確率の z 変換を $P_{ia}(z)$ とすると

$$G_{ia}(z) = (1-z)P_{ia}(z) \quad (4)$$

が成り立つ⁽³⁾。このことから推移確率行列 P において p_{aa} を 0 とおいた行列を P^* とすると $(I - zP^*)^{-1}$ の a 列が $G_{ia}(z)$ ($i = 1, 2, \dots, N$) となる。

ここで,

$$[G_{ij}(z)] = (I - zP^*)^{-1} \quad (5)$$

とする。式 (5) の両辺の左から $(I - zP^*)$ を乗ずると

$$(I - zP^*)[G_{ij}(z)] = I \quad (6)$$

a 列目を取ると

$$(I - zP^*) \begin{bmatrix} G_{1a}(z) \\ G_{2a}(z) \\ \vdots \\ \vdots \\ G_{aa}(z) \\ \vdots \\ \vdots \\ G_{Na}(z) \end{bmatrix} = \sigma_a \quad (7)$$

ただし,

σ_a : a 番目の要素が 1 で他はすべて 0 の列ベクトル

この連立方程式の a 番目の式は次のようになる。

$$\sum_{k=1}^N (\delta_{ak} - p_{ak}z) G_{ka}(z) = 1 \quad (8)$$

ただし,

$$\delta_{ij} = \begin{cases} 1 & ; i=j \\ 0 & ; i \neq j \end{cases} \quad (9)$$

ここで, a は吸収状態であるので $p_{ak} = 0$ ($k \neq a$) であり, また, $p_{aa} = 0$ とおいているので,

$$G_{aa}(z) = 1 \quad (10)$$

が得られる。このことは,

$$(I-zP) \begin{pmatrix} G_{1a}(z) \\ G_{2a}(z) \\ \vdots \\ \vdots \\ G_{aa}(z) \\ \vdots \\ \vdots \\ G_{Na}(z) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix} \quad (11)$$

の連立方程式を作り、 a 番目の式を式 (10) に置き換えた連立方程式を解けばよいことを示している。

ii) 吸収状態の部分集合を入力した場合

状態空間 S から吸収状態の部分集合 A に属する状態を除き、新たにこの集合 A に対応する吸収状態 a を加えた状態空間 S' を考える。この S' におけるマルコフ連鎖において、i) に述べた手法により、 $G_{ia}(z)$ を求めれば、これが元の S におけるマルコフ連鎖の集合 A への吸収時間分布の z 変換 $G_{iA}(z)$ となる。実際のプログラムでは集合 A に対応する状態は状態番号 $N+1$ として扱っている。

3. 3. FPZTRANS

(a) 機能

各状態 i から、実行パラメータで指定された状態 j への初到達時間分布の z 変換 $F_{ij}(z)$ 、又は、状態の集合 C への初到達時間分布の z 変換 $F_{iC}(z)$ を求めて出力する。

(b) 入力パラメータ

(a) に示した状態 j 、又は状態の集合 C を入力する。状態入力か状態の集合入力かはオプション番号により指定する。

• オプション番号

- 1 : 状態を入力することを示す。(リスト形式で複数個の状態の入力が可能)

2 : 状態の集合を入力することを示す。

(c) 生成される REDUCE プログラムにおける計算方法

実行パラメータで状態, 又は状態集合のいずれを指定した場合においても次のような吸収的マルコフ連鎖を形成して計算する。

i) 状態 j を指定した場合

状態空間 S に新たに吸収状態 j^* を加え, 各状態から状態 j への推移を j^* への推移となるようにする。この吸収的マルコフ連鎖における i から j^* までの吸収時間分布の z 変換が $F_{ij}(z)$ となる。

ii) 状態の集合 C を指定した場合

同様に, 状態空間 S に新たに吸収状態 c を加え, 各状態 i から $j \in C$ への推移を c への推移となるようにした吸収的マルコフ連鎖における i から c までの吸収時間分布の z 変換 $F_{ic}(z)$ として求めることができる。

実際のプログラムでは新たに加える吸収状態は状態番号 $N+1$ としている。吸収時間分布の z 変換の求め方は 3. 2 と同様である。

3. 4 ABSORB

(a) 機 能

実行パラメータであるオプション番号に従って, 吸収的マルコフ連鎖の特性量 (訪問回数の期待値, 分散, 吸収時間の期待値, 分散, 及び吸収確率など) を出力する。

(b) 実行パラメータ

出力結果として必要とする特性量を選択するためのオプション番号を入力する。

• オプション番号

- 1 : 訪問回数の期待値の行列と分散の行列, 吸収時間の期待値の行列と分散の行列, 吸収確率の行列を求める。
- 2 : 初期状態を指定して, その初期状態に関する訪問回数の期待値, 吸収時間の期待値を求める。リスト形式で複数個の初期状態の指定が可能である。

3 : 吸収時間の期待値をすべての初期状態に関して求める。

4 : 吸収状態を指定して、その吸収状態に関する吸収確率を求める。リスト形式で複数個の吸収状態の指定が可能である。

(c) 生成される REDUCE プログラムにおける計算方法

吸収的マルコフ連鎖の状態空間 $S = \{1, 2, \dots, m\}$ において吸収状態の集合を $A = \{1, 2, \dots, m-r\}$ とし、 $T = \{m-r+1, m-r+2, \dots, m\}$ を一時的状態の集合とすると、その推移確率行列 P は

$$P = \begin{matrix} A & \begin{pmatrix} I & O \\ R & Q \end{pmatrix} \\ T & \end{matrix} \quad (12)$$

($R \neq O$, I は $(m-r) \times (m-r)$ の単位行列)

の形に書ける。本プログラムで用いる入力ファイル内の推移確率行列はこの形式になっていなければならない。

$i \in T$ が初期状態のとき、いずれかの吸収状態に吸収されるまでの $j \in T$ の訪問回数を n_{ij} , 吸収時間を τ_i , 又、 $i \in T$ から $j \in A$ への吸収確率を b_{ij} とする。各種の特性量は次のような公式で得られることが知られている⁽²⁾。

• 訪問回数の期待値

$$[E\{n_{ij}\}] = N = (I - Q)^{-1} \quad (13)$$

• 訪問回数の分散

$$[Var\{n_{ij}\}] = N(2N_{dg} - I) - N_{sq} \quad (14)$$

• 吸収時間の期待値

$$[E\{\tau_i\}] = N\xi = (I - Q)^{-1}\xi \quad (15)$$

• 吸収時間の分散

$$[Var\{\tau_i\}] = (2N - I)N\xi - (N\xi)_{sq} \quad (16)$$

• 吸収確率

$$B = [b_{ij}] = NR = (I - Q)^{-1}R \quad (17)$$

ただし、

I : $r \times r$ の単位行列

A_{dg} : 行列 A の対角要素を対角要素とする対角行列。

A_{sq} : 行列 A の各要素の 2 乗を要素とする行列。

ξ : すべての要素が 1 であるような列ベクトル。

計算はこれらの公式を元に行うが、実行パラメータのオプション番号により計算方法を変え、効率よく計算できるように工夫している。

i) オプション番号 1 の場合

公式 (13) ~ (17) に従って計算を行う。ただし、逆行列 $N = (I - Q)^{-1}$ は 3.1 にも述べたように REDUCE に組み込まれている逆行列計算法ではあまり大きな行列を扱えないため、次のような連立方程式に変換する。

式 (13) の両辺に右から $(I - Q)$ を掛けると

$$[E\{n_{ij}\}](I - Q) = I \quad (18)$$

行毎に分けると

$$[E\{n_{i1}\}, E\{n_{i2}\}, \dots, E\{n_{ir}\}](I - Q) = \eta_i \quad (19)$$

ただし、

η_i : i 番目の要素が 1 で他はすべて 0 の行ベクトル

$i = 1, 2, \dots, r$ に関してこの連立方程式をそれぞれ解くことにより、行列 $N = [E\{n_{ij}\}]$ を求める。

ii) オプション番号 2 の場合

指定された初期状態 i に関して式 (19) により訪問回数の期待値

$$E\{n_{i1}\}, E\{n_{i2}\}, \dots, E\{n_{ir}\}$$

を求める。又、

$$E\{\tau_i\} = \sum_{j=1}^r E\{n_{ij}\} \quad (20)$$

により、吸収時間の期待値を求める。

iii) オプション番号 3 の場合

式 (15) より

$$(I - Q)[E\{\tau_i\}] = \xi \quad (21)$$

が得られる。この連立方程式により、すべての初期状態に関する吸収時間の期

待値を求める。

iv) オプション番号4の場合

式 (17) より

$$(I-Q)B=R \tag{22}$$

が得られる。 $R=[r_{ij}]$ とし、式 (22) を列毎に分けると

$$(I-Q) \begin{bmatrix} b_{1j} \\ b_{2j} \\ \vdots \\ \vdots \\ \vdots \\ b_{nj} \end{bmatrix} = \begin{bmatrix} r_{1j} \\ r_{2j} \\ \vdots \\ \vdots \\ \vdots \\ r_{nj} \end{bmatrix} \tag{23}$$

となる。これにより、指定された吸収状態 j に関する吸収確率を求める。

3. 5 ERGODIC

(a) 機能

実行パラメータであるオプション番号に従って、エルゴード的マルコフ連鎖の特性量（定常分布，初到達時間の期待値，分散など）を出力する。

(b) 実行パラメータ

オプション番号を入力することにより，出力結果として必要とする特性量を選択する。

• オプション番号

1 : 定常分布，初到達時間の期待値の行列，および初到達時間の分散の行列を求める。

2 : 定常分布と初到達時間の期待値の行列を求める。

3 : 定常分布だけを求める。

(c) 生成される REDUCE プログラムにおける計算方法

i) 定常分布 $\pi = (\pi_1, \pi_2, \dots, \pi_N)$

$$\pi P = \pi \tag{24}$$

$$\sum_i \pi_i = 1 \quad (25)$$

の連立方程式を解くことにより求める。

ii) 初到達時間の期待値の行列と分散の行列

状態 i から j への初到達時間を θ_{ij} で表すと、期待値の行列と分散の行列はそれぞれ $[E\{\theta_{ij}\}]$, $[Var\{\theta_{ij}\}]$ と表される。

$$M = [E\{\theta_{ij}\}] \quad (26)$$

$$W = [E\{\theta_{ij}^2\}] \quad (27)$$

とすると

$$[Var\{\theta_{ij}\}] = W - M_{sq} \quad (28)$$

であり、 M と W は次の公式により得られる^②。

$$M = (I - Z + JZ_{dg})D \quad (29)$$

$$W = M(2Z_{dg}D - I) + 2(ZM - J(ZM)_{dg}) \quad (30)$$

ただし、

J : すべての要素が1の行列

D : $1/\pi_i$ を対角要素とする対角行列で $D = M_{dg}$

$Z = (I - (P - U))^{-1}$

$U = \xi\pi$

ξ : すべての要素が1の列ベクトル

$\pi = (\pi_1, \pi_2, \dots, \pi_N)$: 定常分布

ここで、

$$Z = [z_{ij}] = (I - (P - U))^{-1} \quad (31)$$

は次のような連立方程式に変換して計算する。

式 (31) より

$$Z(I - P + U) = I \quad (32)$$

これを行毎に分けると

$$[z_{i1}, z_{i2}, \dots, z_{iN}] (I - P + U) = \eta_i \quad (33)$$

ただし、

η_i : i 番目が 1 で他は 0 の行ベクトル

$i = 1, 2, \dots, N$ に関してこの連立方程式をそれぞれ解くことにより行列 Z が求まる。

4. 実行例

AZTRANS, ABSORB, 及び, ERGODIC の実行例 (例 1 ~ 例 3) を示す。それぞれの実行例における入力ファイルの内容, 実行時の端末の画面 (実行パラメータ), 生成された REDUCE プログラム, 及び実行結果が図 2 ~ 図 13 に示されている。表 2 はこれらの図と実行例との対応表である。

表 2 各実行例と図 2 ~ 図 13 の対応

	例 1	例 2	例 3
プログラム	AZTRANS	ABSORB	ERGODIC
入力ファイル	図 2	図 6	図 10
実行時の端末の画面	図 3	図 7	図 11
生成された REDUCE プログラム	図 4	図 8	図 12
実行結果	図 5	図 9	図 13

```

*****
* MARKOV CHAIN ANALYZER < VER. 1.0 > *
* ( GET Z-TRANSFORM OF DISTRIBUTION *
* OF TIME TO ABSORPTION ) *
*****
<<< OPTIONS >>>

1 : ENTER ABSORBING STATE LIST
2 : ENTER ABSORBING STATE SUBSET
4
1 0 0 0 * ENTER OPTION !
0 1 0 0 2
1-B 0 0 B * ABSORBING STATE SUBSET ? (END=0)
0 1-A A 0 1 2 0
    
```

図 2 入力ファイル(例1)

図 3 実行時の端末の画面 (例 1)

なお、入力ファイルの最初のレコードは状態数であり、第2レコード以降に推移確率行列が格納される。たとえば、図2では状態数は4であり、推移確率行列 P は次のような形式である。

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1-B & 0 & 0 & B \\ 0 & 1-A & A & 0 \end{bmatrix}$$

```

COMMENT ** Z-TURNFORM OF DISTRIBUTION ;
COMMENT                OF TIME TO ABSORPTION;
COMMENT    < OPTION 2 >;
ON GCD%
FACTOR Z%
SOLVE(LL,LL)%      OFF SOLVEWRITE%
LL:=5%
MATRIX QQ, II(LL,LL), XX, YY, VV, ZZ(LL,1)%
ARRAY SS(2), TT(LL)%
DD:=2%
SS(1):=1% TT(1):=1%
SS(2):=2% TT(2):=1%
BEGIN
WRITE "// ABSORBING STATE SUBSET ";
WRITE "      = (1,2)";
END%
FOR JJ:=1:LL DO II(JJ,JJ):=1%
QQ:=MAT(
(1,0,0,0,0),
(0,1,0,0,0),
(1-B,0,0,0,B),
(0,1-A,A,0,0),
(0,0,0,0,0))%
FOR MM:=1:DD DO BEGIN
KK:=SS(MM)%
FOR JJ:=1:LL-1 DO BEGIN
  QQ(JJ,LL):=QQ(JJ,LL)+QQ(JJ,KK)%
  QQ(JJ,KK):=0%  END%
END%
XX:=MAT((XX1), (XX2), (XX3), (XX4), (XX5))%
YY:=(II-Z*QQ)*XX%
XX5:=1%
WRITE "** Z-TURNFORM OF DISTRIBUTION OF TIME TO ABSORPTION **";
SOLVE(LST(YY(3,1), YY(4,1))
, XX3, XX4)%
KK:=1%
FOR JJ:=1:LL-1 DO
  IF TT(JJ)=0 THEN BEGIN
    WRITE "<", JJ, "> ", A> ", ZZ(JJ,1):=SOLN(1, KK);
    KK:=KK+1%  END%
;END:

```

図4 生成された REDUCE プログラム (例1)

(例1) AZTRANS の実行例

実行パラメータ (図3) でオプション2 (吸収状態の集合入力) を指定し, 吸収状態の集合を $A = \{1, 2\}$ としている。図5の出力において

$$\langle i, A \rangle ZZ(i, 1) := \dots\dots\dots$$

は i から集合 A への吸収時間分布の z 変換 $G_{iA}(z)$ が $ZZ(i, 1)$ に格納されていることを示す。

```
SOLN(1,1) := 0

// ABSORBING STATE SUBSET
= (1,2)

** Z-TRANSFORM OF DISTRIBUTION OF TIME TO ABSORPTION **

<3,A> ZZ(3,1) := (Z^2 * B * (A - 1) + Z * (B - 1)) / (Z^2 * A * B - 1)

<4,A> ZZ(4,1) := (Z^2 * A * (B - 1) + Z * (A - 1)) / (Z^2 * A * B - 1)
```

図5 実行結果 (例1)

```
*****
*           MARKOV CHAIN ANALYZER   < VER. 1.0 > *
*           ( FOR ABSORBING CHAIN )   *
*****
<<<  OPTIONS  >>>

1 : EXPECTED NUMBER OF VISITS TO STATES
   VARIANCE OF THE NUMBER OF VISITS TO STATES
   EXPECTED TIME TO ABSORPTION
   VARIANCE OF THE TIME TO ABSORPTION
   ABSORPTION PROBABILITY

2 : EXPECTED NUMBER OF VISITS TO STATES
   EXPECTED TIME TO ABSORPTION
   ( SPECIFY STARTING STATES )

3 : EXPECTED TIME TO ABSORPTION

4 : ABSORPTION PROBABILITY
   ( SPECIFY ABSORBING STATES )

* ENTER OPTION 1
2
* STARTING STATES ? (END=0)
5 6 0
```

図6 入力ファイル(例2)

図7 実行時の端末の画面 (例2)

```

COMMENT ** ABSORBING MARKOV CHAIN ANALYZER;
COMMENT < OPTION 2 >;
ON GCD¥
SOLVE(LL,LL)¥ OFF SOLVEWRITE¥
LL:=4¥
ZZ:=2¥
MATRIX QQ,II(LL,LL),XX,YY,HH(1,LL)¥
MATRIX NN(2,LL),TT(2,1)¥
ARRAY SS(2)¥
DD:=2¥
SS(1):=3¥
SS(2):=4¥
FOR JJ:=1:LL DO II(JJ,JJ):=1¥
QQ:=MAT(
(B,0,0,0),
(C,B,0,0),
(O,C,B,0),
(O,0,C,B))¥
XX:=MAT((XX1,XX2,XX3,XX4))¥
XX:=XX*(II-QQ)¥
FOR MM:=1:DD DO BEGIN
JJ:=SS(MM)¥
WRITE "// STARTING STATE = ",JJ+ZZ:
HH(1,JJ):=1¥
YY:=XX-HH¥
SOLVE(LST(YY(1,1),YY(1,2),YY(1,3),YY(1,4))
,XX1,XX2,XX3,XX4)¥
WRITE "** EXPECTED NUMBER OF VISITS TO STATES **":
FOR KK:=1:LL DO BEGIN
NN(MM,KK):=SOLN(1,KK)¥
TT(MM,1):=TT(MM,1)+SOLN(1,KK)¥
WRITE "< ",JJ+ZZ," ",KK+ZZ,"> ",NN(" ",MM," ",KK,""):=", NN(MM,KK):
END¥
WRITE "** EXPECTED TIME TO ABSORPTION **":
WRITE "< ",JJ+ZZ,"> ",TT(" ",MM," ",1):=", TT(MM,1):
HH(1,JJ):=0¥
END¥
;END;

```

図8 生成された REDUCE プログラム (例2)

(例2) ABSORB の実行例

実行パラメータ (図7) でオプション2 (初期状態を指定して訪問回数と吸収時間の期待値を求める) を指定し, 初期状態を5と6にしている。図9の出力において

$$\langle i, j \rangle \quad NN(k, m) := \dots\dots\dots$$

$$\langle i \rangle \quad TT(k, 1) := \dots\dots\dots$$

は初期状態が i のときの状態 j の訪問回数の期待値が $NN(k, m)$ に, 状態 i における吸収時間の期待値が $TT(k, 1)$ にそれぞれ格納されていることを示す。


```

SOLN(1,1) := 0

// STARTING STATE = 5
** EXPECTED NUMBER OF VISITS TO STATES **
<5,3> NN(1,1):=( - C2 )/(B3 - 3*B2 + 3*B - 1)
<5,4> NN(1,2):=C/(B2 - 2*B + 1)
<5,5> NN(1,3):=( - 1)/(B - 1)
<5,6> NN(1,4):=0
** EXPECTED TIME TO ABSORPTION **
<5> TT(1,1):=( - B2 + B*C + 2*B - C2 - C - 1)/(B3 - 3*B2 + 3*B - 1)
// STARTING STATE = 6
** EXPECTED NUMBER OF VISITS TO STATES **
<6,3> NN(2,1):=C3/(B4 - 4*B3 + 6*B2 - 4*B + 1)
<6,4> NN(2,2):=( - C2 )/(B3 - 3*B2 + 3*B - 1)
<6,5> NN(2,3):=C/(B2 - 2*B + 1)
<6,6> NN(2,4):=( - 1)/(B - 1)
** EXPECTED TIME TO ABSORPTION **
<6> TT(2,1):=( - B3 + B2*C + 3*B2 - B*C2 - 2*B*C - 3*B + C3 + C2 +
C + 1)/(B4 - 4*B3 + 6*B2 - 4*B + 1)

```

図9 実行結果 (例2)

(例3) ERGODIC の実行例

実行パラメータ (図11) でオプション2を指定し、定常分布と、初到達時間の期待値を求めている。図13の出力結果において定常分布 π_j は UU (1, j) に格納されている。また, i から j への初到達時間分布の期待値 $E \{ \theta_{ij} \}$ は MM (i, j) に格納されている。

```

*****
*           MARKOV CHAIN ANALYZER   < VER. 1.0 > *
*           ( FOR ERGODIC CHAIN )   *
*****
<<<  OPTIONS  >>>

1 : LIMITING STATE PROBABILITIES
   EXPECTED FIRST PASSAGE TIME
   VARIANCE OF FARST PASSAGE TIME

2 : LIMITING STATE PROBABILITIES
   EXPECTED FIRST PASSAGE TIME

3 : LIMITING STATE PROBABILITIES
   * ENTER OPTION !
2
3
1-2*A  A  A
1/2  0  1/2
A  A  1-2*A

```

図10 入力ファイル(例3)

図11 実行時の端末の画面(例3)

```

COMMENT ** ERGODIC MARKOV CHAIN ANALYZER;
COMMENT < OPTION 2 >;
ON GCD¥
SOLVE(LL,LL)¥ OFF SOLVEWRITE¥
LL:=3¥
MATRIX QQ,XX,YY,UU(1,LL)¥
QQ:=MAT(
(1-2*A,A,A),
(1/2,0,1/2),
(A,A,1-2*A))¥
XX:=MAT((XX1,XX2,XX3))¥
YY:=XX*QQ-XX¥
YY(1,LL):=-1¥
FOR JJ:=1:LL DO
  YY(1,LL):=YY(1,LL)+XX(1,JJ)¥
SOLVE(LST(YY(1,1),YY(1,2),YY(1,3))
,XX1,XX2,XX3)¥
FOR JJ:=1:LL DO UU(1,JJ):=SOLN(1,JJ)¥
WRITE "** LIMITING STATE PROBABILITY **";
UU:=UU;
MATRIX HH(1,LL),VV(LL,LL),II(LL,LL),ZZ(LL,LL)¥
MATRIX DD(LL,LL),MM¥
FOR JJ:=1:LL DO II(JJ,JJ):=1¥
VV:=II-QQ¥
FOR JJ:=1:LL DO FOR KK:=1:LL DO
  VV(JJ,KK):=VV(JJ,KK)+UU(1,KK)¥
XX:=XX*VV¥
FOR JJ:=1:LL DO BEGIN
  HH(1,JJ):=1¥
  YY:=XX-HH¥
  SOLVE(LST(YY(1,1),YY(1,2),YY(1,3))
,XX1,XX2,XX3)¥
  FOR KK:=1:LL DO ZZ(JJ,KK):=SOLN(1,KK)¥
  HH(1,JJ):=0¥
  END¥
FOR JJ:=1:LL DO BEGIN
  FOR KK:=1:LL DO VV(JJ,KK):=ZZ(KK,KK)¥
  DD(JJ,JJ):=1/UU(1,JJ)¥
  END¥
WRITE "** EXPECTED FIRST PASSAGE TIME **";
MM:=(II-ZZ+VV)*DD;
;END;

```

図12 生成された REDUCE プログラム(例3)

```

SOLN(1,1) := 0

** LIMITING STATE PROBABILITY **
UU(1,1) := 1/(2*(A + 1))
UU(1,2) := A/(A + 1)
UU(1,3) := 1/(2*(A + 1))

** EXPECTED FIRST PASSAGE TIME **
MM(1,1) := 2*(A + 1)
MM(1,2) := 1/A
MM(1,3) := (2*(A + 1))/(3*A)
MM(2,1) := (4*A + 1)/(3*A)
MM(2,2) := (A + 1)/A
MM(2,3) := (4*A + 1)/(3*A)
MM(3,1) := (2*(A + 1))/(3*A)
MM(3,2) := 1/A
MM(3,3) := 2*(A + 1)

```

図13 実行結果 (例3)

5. む す び

マルコフ連鎖の各種特性量の導出を支援するための、数式処理によるソフトウェア・パッケージ MCA を開発した。これにより、従来、困難であった複雑な問題の特性量を記号パラメータを含む形で導出することが容易となり、システムのパラメータ依存性などの有用なデータを得ることが可能となった。又、数式処理による解は数値解の場合に起こり得る桁落ちや丸め誤差などが生じないため、結果の信頼性が高いという利点もある。

ただし、REDUCE の出力結果においては、何も指定しないと式の各項がすべて展開されてしまうため、欲するような形の出力が得られないことが多い。したがって、得られた結果に対して、適当な REDUCE のフラグ (FACTOR フラグ) をオンにして変数のべきをくくり出したり、FACTORIZE 演算子により

因数分解するなどして、利用者の望むような形に変形する必要がある。

筆者らは、自動生産システムの生産率の解析^{(7),(8)}に MCA の ERGODIC を適用しており、状態数が、文献(7)では18、文献(8)では9の場合の定常分布を得ている。

MCA で扱えるマルコフ連鎖の状態数の上限がどの程度かということは一概には言えない。これは、メモリの大きさに関係し、REDUCE による計算の中間結果がメモリに入りきるかどうかによる。すなわち、単に推移確率行列の大きさ(状態数)だけではなく、どの程度、行列が疎であるか、又、行列の各要素の式がどの程度複雑かにも依存する。この評価はまだ十分ではないが、経験的にはユーザ・メモリが3メガバイトで20程度の状態数の一般のマルコフ連鎖は扱えるのではないかと思われる。状態数の上限の評価は今後の課題としたい。

又、現在、REDUCE はパソコン上でも利用可能であるので⁽⁹⁾、本パッケージの移植を考えたい。

謝辞 本研究を進めるにあたり、ご指導、及び、有益なご助言をいただいた北海道大学工学部情報工学科加地郁夫教授、ならびに大内東助教授に深謝いたします。

文 献

- (1) 森村英典, 高橋幸雄: マルコフ解析, 日科技連 (1979).
- (2) J.G.Kemeny and J.L.Snell: Finite Markov Chains, Van Nostrand (1960).
- (3) R.A.Howard: Dynamic Probabilistic Systems, Vol. 1: Markov Models, Wiley (1971).
- (4) 村尾裕一: “大型数式処理システムの利用の実態”, 情報処理, 27, 4, pp. 371 - 378 (1986 - 04).
- (5) 対馬勝英: “小型数式処理システムとその応用”, 情報処理, 27, 4, pp.

379 - 387 (1986 - 04) .

- (6) A.C.Hearn: REDUCE User's Manual Version 3.2, The Rand Corp., Santa Monica (1985) .
- (7) 中村隆志, 大内 東, 加地郁夫: “マルコフ連鎖によるトランスファー型自動生産システムの解析 — 1 段階 2 機械モデル —”, 電子通信学会論文誌 (A), J 68 - A, 2, pp. 130 - 137 (1985 - 02).
- (8) 中村隆志, 大内 東, 加地郁夫: “マルコフ連鎖による 1 段階トランスファー型自動生産システムの解析 — 2 機械並列システム —”, 電子通信学会論文誌 (A), J 68 - A, 12, pp. 1317 - 1325 (1985 - 12).
- (9) 小林英恒: “Reduce 3.2: 数式処理システム Reduce がパソコンで稼働”, 日経バイト, 31, pp. 121 - 123 (1987 - 04).