

GSM 実現アルゴリズムの一提案

山口 大輔[†]
赤羽根 隆広[‡]

李 国棟[‡]
永井 正武[§]

水谷 晃三[‡]
北岡 正敏[§]

GSM は ISM および FSM に継ぐ新しいシステムモデリング法である。GSM は 2 つの大きな処理からなる、一つは与えられた要素の階層を推定すること、もう一つは与えられた要素間のパスを推定することである。前者は集合演算を含むなど、やや複雑な式から構成されている。本論文では、GSM の一実現アルゴリズムをいくつかの図と共に提案する。特に解決しなければならないのは、与えられた要素をクラスに分類し、そのクラスに階層を与える処理の実現方法である。本論文で提案するアルゴリズムは、誤差行列を解析して要素をクラスに分類する。そして、ミンコフスキー距離型灰色関連度により各クラスに階層情報を与える。提案アルゴリズムは疑似言語で示し、MATLAB によってプログラムを実現した。開発したプログラムを 3 つの例に適用する。

A Realization Algorithm of Grey Structural Modeling

Daisuke YAMAGUCHI[†] Guo-Dong LI[‡] Kozo MIZUTANI[‡]
Takahiro AKABANE[‡] Masatake NAGAI[§] Masatoshi KITAOKA[§]

Grey Structural Modeling (GSM in short) is a new approach of system modeling methods succeeding to ISM and FSM. GSM has two procedures: estimating a hierarchy of given elements, estimating paths among given elements. The former procedure is constructed from complex equations. In this paper we developed one realization algorithm of the GSM procedure with several figures. The main problem we should solve is how to group given elements into several classes and to decide their hierarchy. We are possible to group analyzing an error matrix which is obtained from the localized grey relational grade, and we are also possible to decide their hierarchy according to the localized grey relational grade. We used the topological-based grey relational analysis. These procedures are shown as a pseudo language, and are realized by MATLAB. Several examples applied with the developed program are shown in this paper.

1. はじめに

サイバネティクスでは、与えられた問題の対象をシステムと捉える。システムモデリングは、視覚的にシステムの構成要素を構造化するため、サイバネティクスにおいて重要な手法の一つである。構造化されたグラフは有向グラフ（または構造グラフ）と呼ばれ、各要素の位置を階層で、要素間の関係を矢印（パス）で表す。

これまで Interpretive Structural Modeling (ISM) [1], Fuzzy Structural Modeling (FSM) [2], Flexible ISM [3] などのシステムモデリング法が開発されてきた。筆者らは灰色理論 [7] に基づく GSM [4, 5, 6] と呼ばれる手法を提案している。文献 [6] より、GSM には以下の特徴がある：

1. ISM および FSM と互換性がある

[†]神奈川大学大学院工学研究科, Graduate School of Engineering, Kanagawa University

[‡]帝京大学大学院理工学研究科, Graduate School of Science and Engineering, Teikyo University

[§]神奈川大学工学部, Faculty of Engineering, Kanagawa University

2. 二項関係の値に限らず、任意の観測値を扱うことができる

3. サイクリック構造を回避することができる

GSM は大きく二つの処理からなる：

1. 与えられた要素の階層を推定する

2. 与えられた要素間のパスを推定する

前者の処理は集合演算を含むため、やや複雑である。本論文では GSM を様々な問題に適用するため、上記の処理を実現するアルゴリズムを提案する。

本論文で提案したアルゴリズムは MATLAB によってサンプルプログラムを開発する。MATLAB は多くの科学技術計算命令を備えていること、コンピュータの実験環境として信頼できるプログラムを開発することから本研究で採用した。

以下、2. では GSM の処理を要約し、3. にて疑似言語によるアルゴリズムを提案する。4. および 5. にて計 3 つの例題を示す。

2. GSM 要約

2.1 灰色分析

灰色分析 (GRA)[7, 8, 9] は GSM の基本となる手法であり, 二つの要素間の類似度を灰色関連度として与える。本論文ではミンコフスキー距離型灰色分析 [10] を適用する。他の灰色分析法も適用事例に応じて適用が可能である。

定義 1. 与えられた m 個の要素を $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$ とする。要素 $\mathbf{x}_i (i = 1, 2, \dots, m)$ が n 個の属性を持つとき, 比較ベクトルと定義し以下に示す。

$$\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{in})^\top \quad (1)$$

同様に n 個の属性を持ち, 有向グラフの最上位に配置される要素 \mathbf{x}_0 を参照ベクトルと定義し以下に示す。

$$\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0n})^\top \quad (2)$$

定義 2. 局所型灰色関連度の値を管理するベクトル $\boldsymbol{\gamma}$ を以下に定義する。

$$\boldsymbol{\gamma} = (\gamma_{01}, \gamma_{02}, \dots, \gamma_{0m})^\top \quad (3)$$

ただし,

$$\gamma_{0i} = \frac{\max_{\forall i} \|\mathbf{x}_0 - \mathbf{x}_i\|_\zeta - \|\mathbf{x}_0 - \mathbf{x}_i\|_\zeta}{\max_{\forall i} \|\mathbf{x}_0 - \mathbf{x}_i\|_\zeta - \min_{\forall i} \|\mathbf{x}_0 - \mathbf{x}_i\|_\zeta}, \quad (4)$$

$$\|\mathbf{x}_0 - \mathbf{x}_i\|_\zeta = \sqrt[\zeta]{\sum_{k=1}^n |x_{0k} - x_{ik}|^\zeta}, \zeta \geq 1 \quad (5)$$

定義 3. 全体型灰色関連度の値を管理する行列 Γ を灰色関連マトリクスと呼び, 以下に定義する。

$$\Gamma = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \cdots & \gamma_{1m} \\ \gamma_{21} & \gamma_{22} & \cdots & \gamma_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{m1} & \gamma_{m2} & \cdots & \gamma_{mm} \end{bmatrix} \quad (6)$$

ただし, $i, j = 1, 2, \dots, m$;

$$\gamma_{ij} = 1 - \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_\zeta}{\max_{\forall i} \max_{\forall j} \|\mathbf{x}_i - \mathbf{x}_j\|_\zeta} \quad (7)$$

2.2 階層決定処理

GSM は局所型灰色関連度を参照して, 各要素をいくつかのクラスに分類する。各クラスの階層は灰色関連度の降順整列時の順序に従う。以下にこれらの処理を定義する。

定義 4. 同一階層に配置する要素の集合を灰色階層集合と定義し, C で表す。

$$C_i = \{\mathbf{x}_j | e_{ij} \leq \theta\} \quad (8)$$

ただし, $i, j = 1, 2, \dots, m$; θ は階層係数であり, $0 \leq \theta \leq 1$ で与える。

$$E = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1m} \\ e_{21} & e_{22} & \cdots & e_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ e_{m1} & e_{m2} & \cdots & e_{mm} \end{bmatrix} \quad (9)$$

は誤差行列であり, $e_{ij} = |\gamma_{0i} - \gamma_{0j}|$, $0 \leq e_{ij} \leq 1$ および $e_{ii} = 0$ とする。

定義 5. 以下の二つの条件を満たす集合 C_i (以下 C_{target} と表す) の全要素を有向グラフに配置する。

1. $\text{card}\{C_i\} = \min$
2. $C_i \not\subset C_j$ for all $j, i \neq j$

各クラスの階層は, 与えられた要素の局所型灰色関連度 γ_{0i} の降順に従う。要素が配置されたあと, C_{target} の全ての要素を他の集合 C_j から取り除く。そして, C_{target} 自身の要素も取り除く。これらの処理は, 以下の条件を満たす間続けられる。

$$\bigcup_{i=1}^m C_i \neq \emptyset \quad (10)$$

2.3 パス設定処理

GSM は要素間に矢印, すなわち向きを持つパスを配置する。

定義 6. パスを配置する要素対 $(\mathbf{x}_i, \mathbf{x}_j)$ の集合をパス集合 P と定義し, 以下に示す。

$$P = \{(\mathbf{x}_i, \mathbf{x}_j) | \gamma_{ij} \geq \psi, \gamma_{0i} < \gamma_{0j}\} \quad (11)$$

ただし, ψ はパス係数と呼び, $0 \leq \psi \leq 1$ とする。

定義 7. パス集合 P の全ての要素対 $(\mathbf{x}_i, \mathbf{x}_j)$ について, \mathbf{x}_i から \mathbf{x}_j に向きを持つパスを設定する。ただし, 式 (12) を満たす要素対 $(\mathbf{x}_i, \mathbf{x}_j), (\mathbf{x}_i, \mathbf{x}_k), (\mathbf{x}_k, \mathbf{x}_j)$ (ただし, $i \neq j \neq k$) はパスを省略してもよい。

$$(\psi \leq \gamma_{ij}) \wedge (\psi \leq \gamma_{ik}) \wedge (\psi \leq \gamma_{kj}) \quad (12)$$

3. 実現アルゴリズムの提案

3.1 メインアルゴリズム

文献 [6] より, GSM の処理をフローチャートにより図 1 に示す。階層決定処理およびパス決定処理の擬似言語による実現アルゴリズムをそれぞれ図 2, 図 3 に示す[¶]。図 1 右の処理と図 2 は同じものであり, 図 1 左下の処理と図 3 も同じものである。

3.2 詳細アルゴリズム

階層決定処理には集合演算が含まれているため, パス決定処理よりもやや複雑なアルゴリズムが要求される。本論文では誤差行列の解析により集合演算と同等の処理を実現するアルゴリズムを以下に提案する。

集合演算は C_{target} が定義 5・条件 2 を満たすか判定する処理 (図 2♣), C_{target} の要素を取り除く処理 (図 2♠) で使用されている。各部分の処理を誤差行列解析で実現するアルゴリズムをそれぞれ図 4 および図 5 に示す。

提案アルゴリズムでは長さ m の配列を二つ定義している。

[¶] '%' より後の文はコメントである。

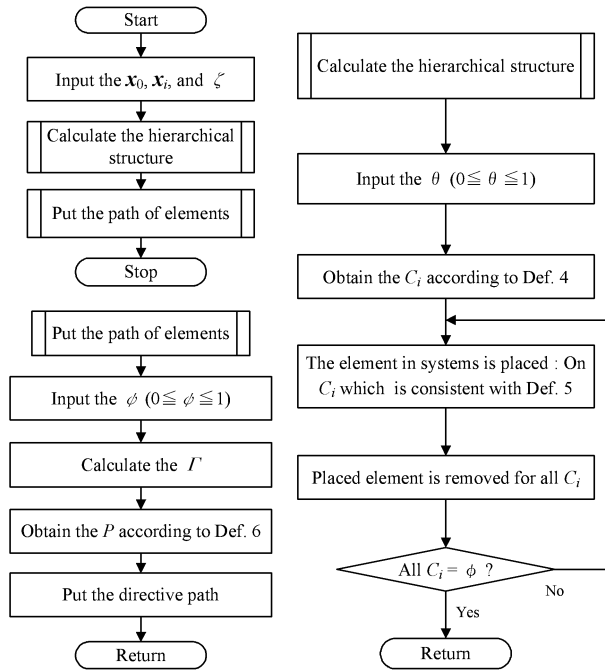


図 1 GSM のフローチャート [6]

ALGORITHM GSM HIEARCHY

$\gamma :=$ Localized GRA $(x_0, x_1, x_2, \dots, x_m, \zeta)$ according to Def. 2;
 Calculating E from γ and θ according to (9) in Def. 4;
 Calculating C from E according to (8) in Def. 4;
 % Grouping process, see Def. 5
 WHILE All of C are NOT an empty set % see (10) in Def. 5
 $C_{\text{target}} := C_i | \exists i, \text{card}\{C_i\} = \min;$ % condition 1) in Def. 5
 % The following is condition 2) in Def. 5, see Fig. 4 (♣)
 flag := Investigate C_{target} is a subset of other $C_i (i \neq \text{target});$
 % Detailed algorithm of the following is shown in Fig. 5 (♠)
 IF flag = TRUE % C_{target} is a subset
 Remove $C_{\text{target}};$
 ELSE % C_{target} is NOT a subset
 Extract all elements of C_{target} and group them into their class;
 Remove extracted elements from other C_i , and remove $C_{\text{target}};$
 Add 1 to the amount of classes;
 END
 END
 Set the hierarchy of each class;

図 2 階層決定処理のアルゴリズム

- elements: $\text{card}\{C_i\}$ を管理する
- class: 要素 x_i が属するクラス番号を管理する

図 4 および図 5 において, e_{ij} の値は 0 または 1 の 2 値になっている。これは式 (8) を

$$e_{ij} = \begin{cases} 1, & \text{if } |\gamma_{0i} - \gamma_{0j}| \leq \theta; \mathbf{x}_j \text{ は } C_i \text{ に属する} \\ 0, & \text{else; } \mathbf{x}_j \text{ は } C_i \text{ に属さない} \end{cases} \quad (13)$$

と展開し, 要素 x_j がクラス C_i に属していることを誤差行列上で管理できるようにするためである。

ALGORITHM GSM PATH

$\gamma :=$ Localized GRA $(x_0, x_1, x_2, \dots, x_m, \zeta)$ according to Def. 2;
 $\Gamma :=$ Globalized GRA $(x_1, x_2, \dots, x_m, \zeta)$ according to Def. 3;
 % Analyzing Γ to find a path, see Def. 6
 FOR $i := 1$ TO m
 FOR $j := 1$ TO m
 IF $(\gamma_{ij} \geq \psi) \wedge (\gamma_{0i} < \gamma_{0j})$ % see (11) in Def. 6
 % Set the directive path between instances x_i and x_j
 Put the path from x_i to x_j ;
 IF Digraph is NOT simplified % see Def. 7
 FOR $k := 1$ TO m
 flag := Investigate the pairs $(x_i, x_k), (x_k, x_j), (x_i, x_j)$ are
 the grey transitive pair according to (12);
 IF flag = TRUE % the pairs are the grey transitive pair
 Put the path from x_i to x_k ;
 Put the path from x_k to x_j ;
 END
 END
 END
 END
 END
 END

図 3 パス決定処理のアルゴリズム

ALGORITHM GSM ANALYZE ERROR MATRIX (♣)

% flag1 = TRUE $\Leftrightarrow C_{\text{target}}$ is a subset (result);
 % flag1 = FALSE $\Leftrightarrow C_{\text{target}}$ is NOT a subset (result);
 % flag2 = TRUE $\Leftrightarrow C_{\text{target}} \not\subset C_i (i \neq \text{target});$
 % flag2 = FALSE $\Leftrightarrow C_{\text{target}} \subset C_i (i \neq \text{target});$
 flag1 := FALSE;
 FOR $i := 1$ TO m
 IF $(i \neq \text{target}) \wedge (\text{elements}(i) \neq m + 1)$
 flag2 := FALSE;
 FOR $j := 1$ TO m
 IF $(e_{\text{target},j} = 1) \wedge (e_{ij} = 0)$
 flag2 := TRUE;
 END
 END
 IF flag2 = FALSE
 flag1 := TRUE;
 END
 END
 END
 RETURN flag1;

図 4 C_{target} の条件判定アルゴリズム

集合 C_{target} は表 1 に示されている論理式によって, 誤差行列上で定義 5・条件 2 の判定を行う。アルゴリズム簡略化のため, 上から 3 番目の論理式を提案アルゴリズムに導入している。要素 x_j は配列 $\text{class}(j)$ にクラス番号を与え, $e_{ij} = 0$ とすることで集合から取り除かれる。全要素が取り除かれた集合 C_{target} は配列 $\text{elements}(\text{target}) = m + 1$ とすることで取り除かれる。

階層情報はクラスごとに属している要素の局部型灰色関連度で平均値を算出し, それらの降順に従う。

ALGORITHM GSM ANALYZE ERROR MATRIX (♠)

```

% flag1 = TRUE ⇔ C_target is a subset (result);
% flag1 = FALSE ⇔ C_target is NOT a subset (result);
classnumber := 1;
IF flag1 = TRUE
    % Remove C_target
    FOR j := 1 TO m
        e_target,j := 0;
    END
    elements(target) := m + 1;
ELSE
    FOR j := 1 TO m
        IF e_target,j = 1
            % Extract elements of C_target and group them into their class
            class(j) := classnumber;
            % Remove extracted elements from other C_i
            FOR i := 1 TO m
                IF e_ij = 1
                    e_ij := 0;
                    IF elements(i) ≠ m + 1
                        elements(i) := elements(i) - 1;
                    END
                END
            END
        END
    END
    % Remove C_target
    elements(target) := m + 1;
    % Add 1 to the amount of classes
    classnumber := classnumber + 1;
END

```

図 5 C_{target} の要素の分類および消去アルゴリズム

表 1 部分集合判定のための論理式

$e_{target,j}$	e_{ij}	判定
0	0	$C_{target} \subset C_i$
0	1	$C_{target} \subset C_i$
1	0	$C_{target} \not\subset C_i$
1	1	$C_{target} \subset C_i$

$target \neq i; j = 1, 2, \dots, m$

3.3 MATLAB による開発

上記の提案アルゴリズムを MATLAB^{||}によって、サンプル GSM プログラムを開発した。図 6 に動作例を示す。利用者は 3 つのパラメータ ζ, ψ および θ を画面上で入力することができる。同様に画面上のチェックボックスにて有向グラフの簡略化、言い換えると灰色推移律を満たすパスを省略することができる。

4. 例題 1

4.1 動作検証

最初の例は、開発したプログラムの動作を確認するため、表 2[6] に示すサンプルシステムに GSM を適用する。このサンプルシステムは、意味を持

^{||}MATLAB Version 7.1.0.246 (R14) SP3.

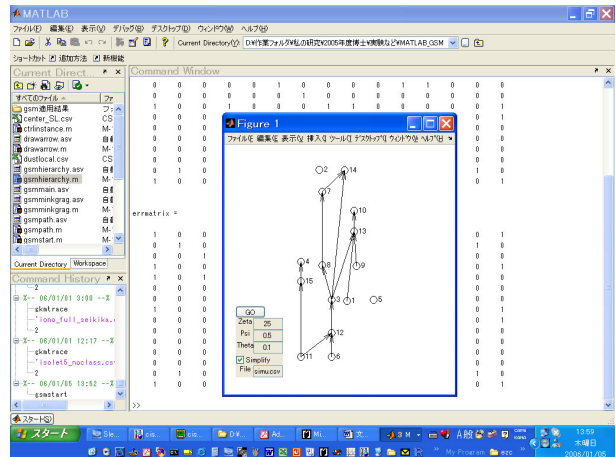


図 6 MATLAB にて開発した GSM プログラム

表 2 サンプルシステムデータ [6]

elements	attributes			
	x_0	1	1	1
x_1	0.645	0.118	0.515	0.233
x_2	0.898	0.368	0.842	0.940
x_3	0.079	0.293	0.647	0.409
x_4	0.289	0.797	0.143	0.655
x_5	0.332	0.839	0.182	0.097
x_6	0.110	0.018	0.863	0.467
x_7	0.666	0.811	0.984	0.278
x_8	0.463	0.775	0.709	0.143
x_9	0.944	0.626	0.166	0.317
x_{10}	0.911	0.499	0.222	0.699
x_{11}	0.019	0.580	0.292	0.798
x_{12}	0.037	0.127	0.459	0.696
x_{13}	0.567	0.277	0.215	0.408
x_{14}	0.356	0.657	0.759	0.758
x_{15}	0.243	0.996	0.141	0.759

Data from M. Nagai et al., *Journal of Grey System*, vol.8, no.2, pp.119–130 (2005).

たない人工的なデータとして 4 属性 15 要素および参照ベクトルからなっている。3 つのパラメータは $\zeta = 2, \psi = 0.5$ および $\theta = 0.1$ とする。

図 7 は上記の入力に対する有向グラフである。そしてこのグラフは図 8 に簡略化することができる。本論文の二つのグラフと文献 [6] のグラフは同じ結果となっている。

4.2 パラメータによる有向グラフの構造変化

有向グラフは 3 つのパラメータにより柔軟にその構造を変えることができる。弁別係数 ζ は有向グラフの基本構造を調節することができる。例えば $\zeta = 25, \psi = 0.5, \theta = 0.1$ としたときの有向グラフを図 9 に示す。他のパラメータ ψ, θ は同じ値である。パス係数はパスの総数を調節することができる。

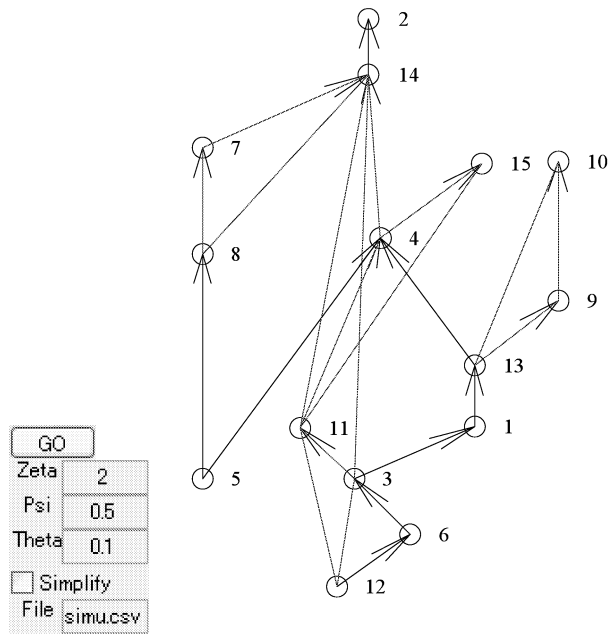


図 7 サンプルシステムの有向グラフ

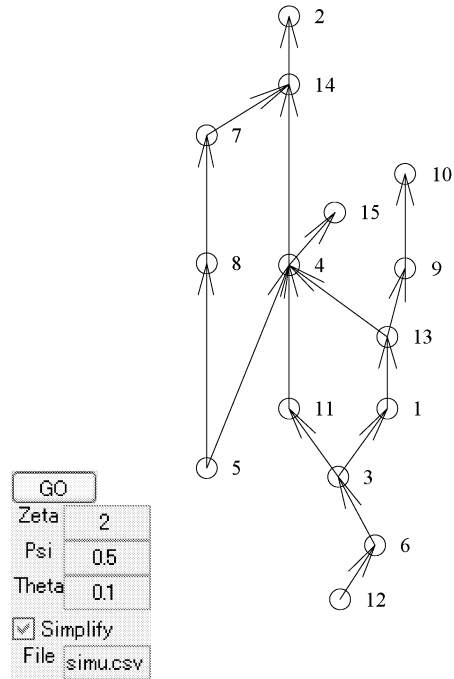


図 8 サンプルシステムの簡略化した有向グラフ

一般的に $\psi \rightarrow 1$ になるにつれてその総数は減り、 $\psi \rightarrow 0$ となるほどその数が増える。例として $\zeta = 2, \psi = 0.6$ および $\theta = 0.1$ としたときの有向グラフを図 10 に示す。図中の 3 つのサブ有向グラフはパラメータの値を変えることで初めて見つけることができる。

クラス係数 θ は、階層的構造を変化させる。値が増加するにつれて階層数は減少し、最終的には一階層、すなわち全ての要素が横一列に配置される。また $\theta \rightarrow 0$ となるにつれて階層数は増加し、最終的には階層数と要素数が等しくなる。本適用例では $\theta = 0.3$ のとき 6 から 2 階層になった。そして、 $\theta = 0$ のとき階層数は 15 となった。ただし、他のパラメータは $\zeta = 2, \psi = 0.5$ で固定している。

5. 追加例題

5.1 例題 2

筆者ら [11, 12, 13] は学生が効率的に学習可能なシステムを開発している。学習項目を構造化によって学生に効率的な学習順序を与えることは、教育学において伝統的な課題の一つである。ISM や FSM ではすでにこれらの問題に適用されている。本論文では 2 番目の例として、学習項目構造化を GSM で行う。

表 3[5] は、プログラミング言語の学習を例とした、各学習項目に費やした時間の総学習時間に対する比率である。学生 16 名が 18 項目の学習時間を測定し、その比率を計算した。参照ベクトルは表の t_0 を用い、各学習項目の学習順序は有向グラフとして整理される。

図 11 が表 3 のデータに対する基本的な有向グラフである。ただし、この図はパスが多いので、パラ

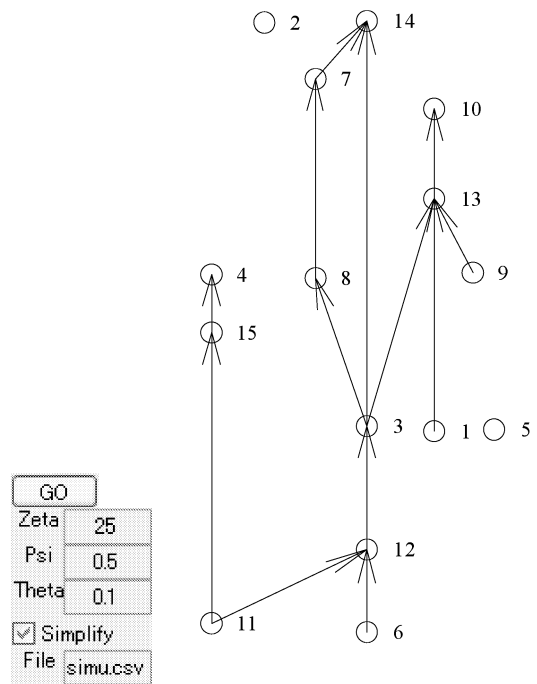


図 9 弁別係数を変えたときの有向グラフの例

メータを調節して図 12 に簡略化する。図 12 から二つの学習経路を見つけることができる。一つは学生の学習環境に関するもの (16 \rightarrow 15) で、もう一つはプログラミング言語習得に最小限必要と思われる学習項目 (18 \rightarrow 10) である。

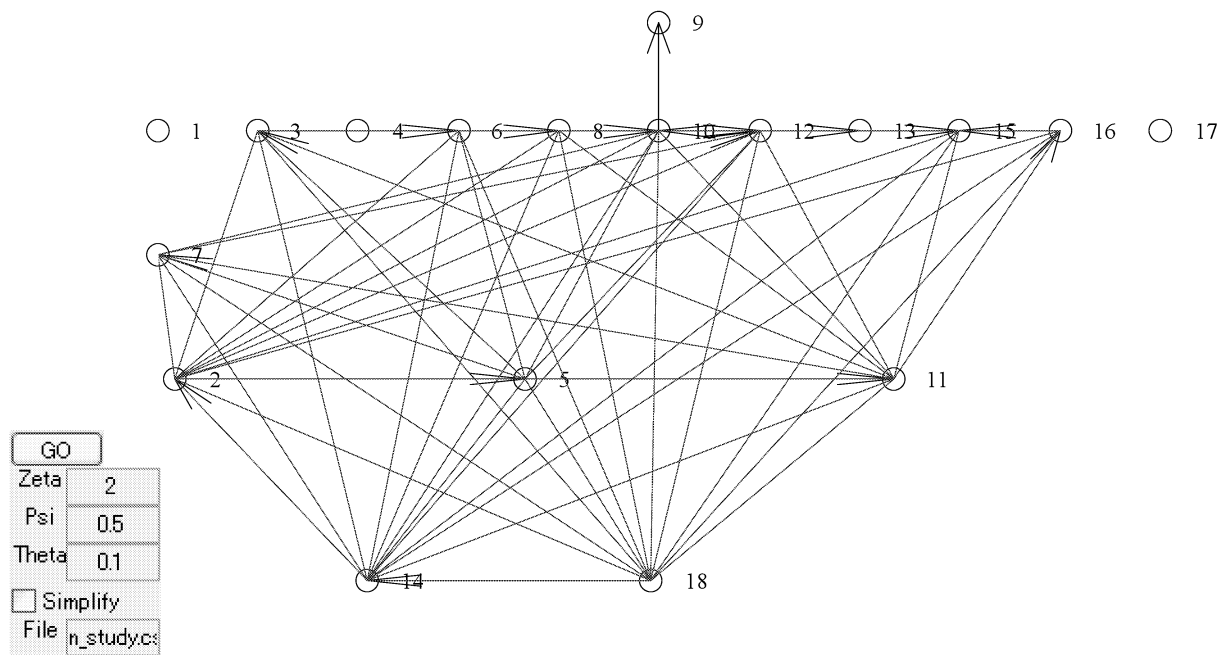


図 11 学習データの有向グラフ

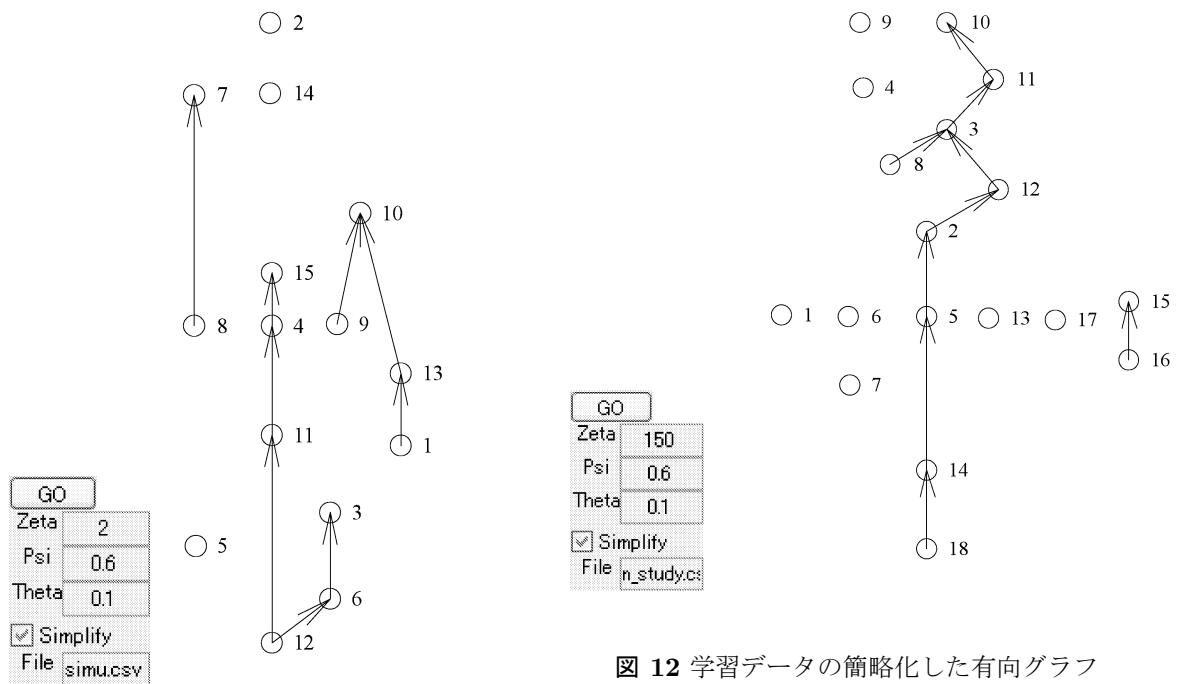


図 12 学習データの簡略化した有向グラフ

図 10 パス係数を変えたときの有向グラフの例

5.2 例題 3

本適用例 [14] は 38 の陶磁器が作られた時代を 22 の元素の含有量から分析する。文献 [14] ではファジークラスタリングなどで分類を行っている。本論文では 3 番目の例として GSM による時代推定と分

類を行う**。

各陶磁器は 4 つの王朝，唐 (Tang)・元 (Yuan)・宋 (Song)・現代 (Today) に分かれており，38 の陶磁器のうち 1 つは現代のものである。現代の陶磁器を参照ベクトルとして有向グラフを作成し，各陶磁器がいつの時代のものかを構造的に調べる。

**本適用例は文献 [6] にて一度適用している。適用結果の詳細は文献 [6]，データは文献 [14] に示されている。

表 3 プログラミング言語の総学習時間に対する 18 項目の学習比率 [5]

	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	s_{16}
t_0	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
t_1	2.3	0.0	0.0	86.3	0.0	0.0	2.0	3.6	5.0	0.0	0.0	12.7	0.0	22.6	0.0	0.0
t_2	0.6	0.0	0.0	2.2	20.0	0.5	0.0	3.4	1.5	0.0	0.0	0.7	5.5	6.2	0.0	0.7
t_3	3.5	0.0	0.0	1.5	9.8	13.7	0.0	21.3	1.4	0.0	0.0	0.9	1.3	31.8	0.0	1.7
t_4	4.1	3.4	0.0	1.6	18.9	1.9	0.0	0.0	0.0	0.0	0.0	75.8	12.5	7.3	0.0	1.0
t_5	0.6	0.0	0.0	2.1	29.2	2.8	0.0	0.0	0.0	0.0	0.0	0.6	5.2	4.3	0.0	0.0
t_6	44.9	0.0	0.0	3.8	22.0	4.8	0.0	0.0	0.0	0.0	0.0	1.9	4.9	27.8	0.0	0.0
t_7	1.3	0.0	0.0	0.0	0.0	0.7	9.6	0.0	4.0	0.0	0.0	0.0	42.5	0.0	7.9	0.0
t_8	0.0	9.8	0.0	0.0	0.0	3.8	2.1	46.9	6.9	0.6	7.1	0.0	12.9	0.0	0.0	12.1
t_9	5.3	0.0	12.3	0.0	0.0	3.0	14.7	2.1	37.1	45.5	7.6	4.7	0.7	0.0	42.2	0.6
t_{10}	1.0	8.6	14.0	0.0	0.0	6.0	3.5	5.9	37.5	5.9	0.0	0.6	6.9	0.0	11.4	0.7
t_{11}	0.8	0.0	13.8	2.6	0.0	6.4	1.1	0.0	3.1	2.4	0.0	0.7	1.8	0.0	9.0	0.7
t_{12}	1.3	0.0	30.7	0.0	0.0	5.2	15.3	0.0	2.4	0.0	0.0	0.6	1.5	0.0	8.3	22.6
t_{13}	0.0	0.0	29.1	0.0	0.0	2.5	51.8	0.0	1.1	0.0	0.0	0.8	4.2	0.0	5.3	34.0
t_{14}	3.5	0.0	0.0	0.0	0.0	1.4	0.0	0.0	0.0	0.0	1.4	0.0	0.0	0.0	0.0	0.0
t_{15}	2.9	36.6	0.0	0.0	0.0	22.9	0.0	2.0	0.0	0.0	18.9	0.0	0.0	0.0	13.5	13.7
t_{16}	26.7	41.7	0.0	0.0	0.0	17.8	0.0	5.3	0.0	0.0	10.8	0.0	0.0	0.0	2.5	0.9
t_{17}	1.2	0.0	0.0	0.0	0.0	2.2	0.0	9.6	0.0	45.7	54.2	0.0	0.0	0.0	0.0	11.1
t_{18}	0.0	0.0	0.0	0.0	0.0	4.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

t_1 プログラミング言語とは何か t_7 Visual Basic の基礎 t_{13} プログラムの実行方法
 t_2 プログラム生成の流れ t_8 Visual Basic の起動方法 t_{14} 高級言語とは何か
 t_3 主な高級言語 t_9 フォームの作成方法 t_{15} エディタの使い方
 t_4 基本命令体系 t_{10} ツールボックスの利用方法 t_{16} コンパイル方法
 t_5 基本処理命令 t_{11} 動作の指定方法 t_{17} 実行コマンド
 t_6 アルゴリズムとは何か t_{12} ソースコードの記述 t_{18} 機械語とは何か

得られた有向グラフのうちの一つを図 13 に示す。有向グラフは 5 階層で構成されており、点線で囲まれている陶磁器はそれぞれ同一時代のものであることを表す。図 13 に示すグラフは文献 [6] にて適用したグラフと同じ結果となっている。

6. まとめ

本論文では GSM の処理を実現するアルゴリズムをフローチャートおよび擬似言語を用いて提案した。また、MATLAB によりサンプルプログラムを開発し、3 つの例題を通じて動作検証を行った。

GSM は二項関係の値だけでなく任意の観測値から構造化が可能である。広範囲な分野で適用が期待されるため、今後さらに多くの実例を示す必要がある。

参考文献

- [1] J.N. Warfield, Binary Matrices in System Modeling, *IEEE Trans. Syst. Man. and Cybern.*, vol.SMC-3, No.5, 1973, pp.441-449.
- [2] E. Tazaki and M. Amagasa, Structural Modeling in a Class of Systems Using Fuzzy Sets Theory, *Fuzzy Sets and Systems*, vol.2, 1979, pp.87-103.
- [3] A. Ohuchi and I. Kaji, Correction Procedures for Flexible Interpretive Structural Modeling, *IEEE Trans. Syst. Man. and Cybern.*, vol.19, no.1, 1989, pp.85-94.
- [4] D. Yamaguchi, T. Kobayashi, K. Mizutani and M. Nagai, Proposal of System Modeling Method Based on Grey Theory, *IPSJ SIG Technical Report*, vol.2004, no.76, AL-96-4, 2004, pp.25-32(in Japanese).
- [5] M. Nagai, D. Yamaguchi, G.D. Li and K. Mizutani, New System Modeling Method Based on Grey Theory, *IPSJ SIG Technical Report*, vol.2005, no.40, AL-101-9, 2005, pp.59-66 (in Japanese).
- [6] M. Nagai, D. Yamaguchi and G.D. Li, Grey Structural Modeling, *Journal of Grey System*, vol.8, no.2, 2005, pp.119-130.
- [7] J.L. Deng, *Grey Systems*, China Ocean Press, 1988.
- [8] M. Nagai and D. Yamaguchi, *Elements on Grey System Theory and Applications*, Kyoritsu Shuppan, 2004 (in Japanese).
- [9] K.L. Wen, *Grey Systems: Modeling and Prediction*, Yang's Scientific Research Institute, 2004.
- [10] D. Yamaguchi, G.D. Li and M. Nagai, New Grey Relational Analysis for Finding the Invariable Structure and its Applications, *Journal of Grey System*, vol.8, no.2, 2005, pp.167-178.
- [11] 水谷晃三, 池本 悟, 山口大輔, 永井正武, "MAS-KNOPPIX による Web 型学習システムの提案," 信学技報, May 2005.

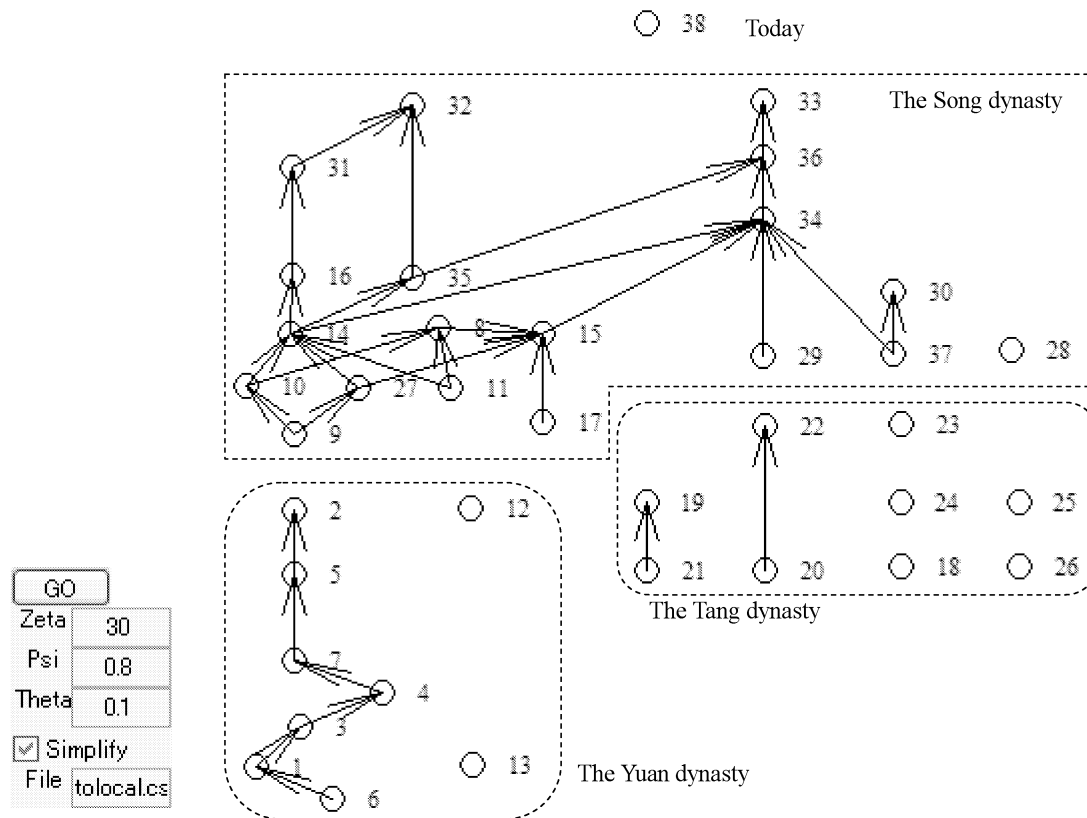


図 13 中国陶磁器の有向グラフによる構造解析結果

- [12] 池本 悟, 小林俊裕, 水谷晃三, 永井正武, “学習経路提示とグループ化による Web 教育への実践と考察,” 信学技報, May 2005.
- [13] B.Y. Cai, M. Nagai, J.R. Zhong, The Comparison and Development Strategy of Web-Based Learning And Traditional Learning by 5W1H Method and Interpretive Structure Model, *GCCCE/NEIT 2002*, pp.198-205.
- [14] A. Wan, P. Wu, S. Liu, and J.M. Miao, An archaeological Judgments of Old Chinawares Based on Neutron Analysis Method and Fuzzy Cluster Analysis, *Beijing Univ. Bulletin*, vol.28, no.6, 1992 (in Chinese).