北海道大学工学部研究報告
第138号（昭和63年）

Bulletin of the Faculty of Engineering,
Hokkaido University. No. 138(1988)

弱連結成分抽出によるグラフ理論的クラスター分析アルゴリズムの検討

大内　東*，加地　郁夫*，栗原　浩一**

（昭和62年9月30日受理）

Graph-theoretic Cluster Analysis Algorithms Based on the Concept of Weak Connected Components of a Digraph.

Azuma OHUCHI*, and Ikuo KAJI* and Khoichi KURIHARA**

(Received September 30, 1987)

Abstract

A sequence of digraphs can be associated with a given matrix by converting it into binary matrices. Any such binary square matrices can be interpreted as an adjacent matrix of a digraph in which there is a directed edge from vertex i to vertex j, iff the (i, j) entry of the binary matrix is 1. The vertices of any digraph can be partitioned into weak components. By taking the thresholds in the ascending order, and determining the weak components of each of the associated digraphs, vertices can be hierarchically clustered. Ten algorithms for the clustering are implemented and their efficiency is compared.

1．まえがき

システムを構成している要素集合を何らかの基準によっていくつかのグループに分類して，システムの構造を知ろうとする手法に，クラスター分析がある。クラスター分析は多くの分野で利用されている[1]。

本論文では，システム構成要素間の一般には非対称行列である類似度（あるいは非類似度）行列を入力データとし，類似度行列をあるしきい値で2値化し，対応する有向グラフの弱連結成分を1つのクラスターと見なすグラフ理論的クラスター分析のアルゴリズムについて検討する。この方法においては，分析者がコンピュータを対話形式で利用し，次々としきい値を変えて，結果を検討しながら実行することが多いため，速やかな応答が要求される。応答時間は，各しきい値毎にそのしきい値における有向グラフの弱連結成分を抽出するアルゴリズムに依存する。また，弱連結成分の抽出を効率良く実行するための前処理も重要である。そこで，著者は10のアルゴリズムをインプリメントし，それらの処理時間を理論と実験の両面から比較を行ない，対話型グラフ理論的クラスター分析に適しているアルゴリズムの検討を行なった。

＊　情報工学科システム工学講座
＊＊　日本鉱業株式会社

2．諸 定 義

以下で，G(V, D) は有向グラフを，G(V, E) は無向グラフを表わす。ここで，Vは有向（無効）グラフを構成する頂点の集合を表わし，Dは有向辺の集合，Eは無向辺の集合を表わす。

グラフの用語（例えば，"準路"，"路"，"路の長さ" 等）に関する定義については標準的定義に従う[8)9)]。

［定義1］類似度行列

類似度行列Fとは，データを構成する要素集合をその添字集合とし，その第(i−j)成分を $0 \leqq f_{ij} \leqq 1$ として要素iとjの間の非負類似度をもつn次正方行列のことをいう。$f_{ii}=1$，for all i, である。

［定義］弱連結

有向グラフにおいて，すべての二頂点間に準路が存在するとき，その有向グラフは弱連結グラフであるといい，任意の有向グラフについて，その極大な弱連結部分グラフを，その有向グラフの弱連結成分という。

［定義3］隣接行列と可到達行列

G(V, D) において，行列の行（列）の添字集合がグラフの頂点集合Vに等しく，$(v_i, v_j) \in D$ のとき，かつその時に限り $a_{ij}=1$ とし，それ以外の時には $a_{ij}=0$ とした行列 $A=\{a_{ij}\}$ を，グラフGの隣接行列という。また，やはり行列の行（列）の添字集合は，グラフの頂点集合Vに等しく，頂点 v_i から頂点 v_j への（長さ0以上の）有向路が存在する時かつその時に限り $a_{ij}=1$ とし，それ以外の時には $a_{ij}=0$ とした行列 $A^*=\{a_{ij}\}$ をグラフGの可到達行列という。ただし，各頂点から自分自身への長さ0の有向路が存在するとする。A^*は $(A+I)$ の推移的閉包として求められる。すなわち，

$$A^*=I+A^1+A^2+\cdots\cdots+A^n,$$

ただし，Iは単位行列である。

無向グラフ G(V, E) に対しても同様に定義する。無向グラフの隣接行列，可到達行列を B, B^*で表わす。B, B^*は対称行列となる。

［定義4］行列Fのtによる二値化

$f_{ij} \geqq t$ ならば $a_{ij}=1$，そうでなければ $a_{ij}=0$ として，行列Fから二値行列 A_t を求める。これを，行列Fのtによる二値化と呼ぶ。A_t には A_t を隣接行列とする有向グラフが一意に対応する。

［定義5］弱連結成分抽出によるクラスター分析

以上の定義をもちいると，類似度行列を用いた弱連結成分抽出によるクラスター分析は，コンピュータとの対話形式で以下のように実行される。

［階層的クラスター分析の過程］

1．Fを入力する。
2．しきい値tを決定する。
3．弱連結成分を探索するための前処理を行なう。
4．しきい値tにおける有向グラフの中で，同一の弱連結成分を構成する頂点集合を一つのクラスターとする。
5．クラスターの数Nが，$N \geqq 2$ ならばステップ2へ行く。$N=1$ ならば終了する。

3．前　処　理

しきい値tに於けるグラフを表わすデータ構造として隣接行列あるいは可到達行列である二値行列を利用する。このため，要素が実数であるn次正方行列Fから隣接行列あるいは可到達行列を生成するための前処理が必要となる。これらの前処理には以下の処理がある。

(1) Fのsによる二値化により A_s を生成する。
(2) A_s から対称行列 B_s を求める。
(3) 二値行列 $A_s(B_s)$ の推移的閉包 $A_s^*(B_s^*)$ を計算する。
(4) $A_s^*(B_s^*)$ の変化に伴う新たな $A_t^*(B_t^*)$ を計算する。
(5) Fから超計量不等式を満たす行列 F^* を計算する。

(1) は定義から容易に実行できる。

(2) は $B_s=A_s+(A_s)^T$（A_sの転置行列）を実行する。有向グラフ $G_s(V, D)$ において，$(v_i, v_j) \in D$ or $(v_j, v_i) \in D$ ならば無向グラフ $G_s'(V, E)$ において $(v_i, v_j) \in E$ とすると，準路の定義から有向グラフ G_s の弱連結成分は無向グラフ G_s' の連結成分に等しい。B_s は G_s' の隣接行列に対応する。B_s を使うことにより弱連結成分の探索が効率良く実行できる。

(3) の計算には後述するWarshallのアルゴリズムを適用する。

(4) の意味は，隣接行列 $A_s(B_s)$ とその推移的閉包 $A_s^*(B_s^*)$ を持つとき，$A_s^* \leqq A_t^*(B_s^* \leqq B_t^*)$ なる $A_s^*(B_s^*)$ の推移的閉包 $A_s^*(B_s^*)$ を $A_s^*(B_s^*)$ を利用して計算することである。ここで，$A_s^* \leqq A_t^*(B_s^* \leqq B_t^*)$ の意味は，$(A_s)^*_{ij} \leqq (A_t)^*_{ij}((B_s)^*_{ij} \leqq (B_t)^*_{ij})$，for all i, j,である。$A_s^*(B_s^*)$ で0から1へ値が変化した要素数が少ないときはWarshallのアルゴリズムを適用しなくても効率良く $A_s^*(B_s^*)$ を計算することができる。

(5) の F^*の計算アルゴリズムとしてBoydアルゴリズムがある[3]。しかし，このアルゴリズムは非負実数行列に対する一般化Warshallアルゴリズムである事が示されている[4]。

3．1　推移的閉包

一般にQ半環と呼ばれる代数系$<Q, +, *, 0, 1>$[6]の元を要素とするn次正方行列の推移的閉包は，一般化Warshallアルゴリズムにより $O(n^3)$ で計算できる[5]。

2値ブール代数$<Q=\{0, 1\}, +, \cdots, 0, 1>$はQ半環である。このとき，Qの元を要素とする一般には非対称な2値行列の推移的閉包は $O(n^3)$ で計算できる[2][8][9]。更に，Qが2値対称行列の場合には，$O(n^2)$ で計算できる事が知られている。

3．2　推移的閉包の部分修正

次に(4)の前処理について考察する。今，可到達行列 A^* において，$a_{ij}^*=0$ なる要素が，$a_{ij}^*=1$ と変化したと仮定する。A^* に対応する有向グラフ G^* で考えると，頂点iから頂点jへの有向辺が新たに加わったことになる。この場合，頂点iへ到達可能なすべての頂点から，頂点jから到達可能なすべての頂点への有向路が新たに存在することになる。A^* に対しこれを実行するには次のようにする。

［手続き　PB-Warshall］
{推移的閉包 A^* の修正。新たに1となったすべての要素 a^*_{ij} に対し以下の計算を実行する}
1．begin

```
2．  for p=1 to n do
3．  if a*pi=1 then
4．    for q=1 to n do
5．    a*pq ：=a*pi+a*jq ；
6．end
```

このアルゴリズムはWarshallアルゴリズムを第j列に対してのみ適用した結果になっている。

同様に，無向グラフの可到達行列B^*において，$b_{ij}^*=0$なる要素が，$b_{ij}^*=1$と変化したと仮定する。B^*に対応する無向グラフG^*で考えると，頂点iと頂点jの間に無向辺が新たに加わったことになる。この場合，頂点iが属する連結部分グラフを構成するすべての頂点($b^*_{ip}=1$なるp)と，頂点jが属する連結部分グラフを構成するすべての頂点($b^*_{jq}=1$なるq)の間に無向辺が新たに存在する($b^*_{pq}=b^*_{qp}=1$)ことになる。従って，B^*のi行とj行のブール和をとり，$b^*_{ip}=1$なるp行と$b^*_{jq}=1$なるq行へコピーする。これにより，B^*の部分修正が終了する。アルゴリズムとして表わすと次のようになる。

```
［手続き　PSB-Warshall］
{推移的閉包B*の修正。新たに1となったすべての要素b*ijに対し以下の計算を実行する}
1．begin
2．  for q ： =1 to n do
3．    b*iq ： =b*iq+b*jq ；
4．  for p ： =1 to n do
5．    if b*ip=1 then
6．      for q ： =1 to n do b*pq ： =b*iq ；
7．end
```

このアルゴリズムは対称2値行列に対するWarshallアルゴリズムを第i行に対してのみ適用した結果になっている。

3．3　超計量行列

類似度行列Fの超計量行列F^*は，Q半環<Q=[0，1]，max，min，1，0>に対する一般化Warshallアルゴリズムにより計算できる。(5)の計算にはこのアルゴリズムを適用する。Fの推移的閉包行列F^*はどんなしきい値について二値化しても，行列Fを同じしきい値で二値化した行列A_tの閉包行列A_t^*に等しくなる。

4．弱連結成分探索のアルゴリズム

弱連結成分探索のアルゴリズムは大きく，

(1)　隣接行列を用いるアルゴリズム

(2)　閉包行列を用いるアルゴリズム

の二つに分けられる。

4．1　隣接行列を用いるアルゴリズム

隣接行列を用いるアルゴリズムとは，隣接行列A（隣接行列B）で表わされている有向グラフG(V, D)（無向グラフG'(V, E)）の中の有向辺（無向辺）を深さ優先探索ですべてたどりながら，弱連結成分を構成する頂点を求めるものである。再帰的手続きを用いたアルゴリズムは文献(9)に述べられている。

4．2　隣接行列修正アルゴリズム

しきい値を常にs<tとなるように選ぶと，$A_s \leqq A_t$ である。今，有向グラフGにおいて，頂点iから頂点jへの有向辺が新たに加わったとする。頂点iと頂点jが同一の弱連結成分に属している場合には，ひとつ前のしきい値で求められたクラスターについての変更を行なう必要はない。一方，頂点iと頂点jが異なる弱連結成分に属している場合には，頂点iの属する弱連結成分を構成するすべての頂点と，頂点jの属する弱連結成分を構成するすべての頂点とが，新たにひとつの弱連結成分を構成することになる。従って，頂点iの属する弱連結成分と，頂点jの属する弱連結成分を新たにひとつのクラスターとすれば部分的な変更だけによって新たなクラスターを求めることができる。

$A_s \leqq A_t$ から $B_s \leqq B_t$ であるので，上と同様にして，B_t に対する修正アルゴリズムを構成することができる。

4．3　可到達行列を用いるアルゴリズム

弱連結の定義からある頂点iから可到達な頂点集合とiへ可到達な頂点集合はすべて同一の弱連結成分となる。そこで可到達行列A*を求めておくことにより弱連結成分の探索効率をあげることが考えられる。可到達行列A*から弱連結成分を構成する要素を捜し出す手続CDA-Searchは以下の様に記述できる。

```
[手続き CDA-Search]
{可到達行列 A* から弱連結成分を求める手続き}
 1．for i=1 to n do
 2．  if (i はどの弱連結成分にも含まれていない)
 3．    then
 4．      begin
 5．        for i=1 to n do
 6．          if a*ij = 1
 7．            then (要素 i と要素 j は新たな弱連
 8．                  結成分を構成する)
 9．        for j=1 to n do
10．        if a*ji = 1
11．          then (要素 i と要素 j は新たな弱連
12．                結成分を構成する)
13．      end
```

無向グラフの可到達行列B*から弱連結成分を構成する要素を捜し出す手続CDB-Searchは

以下の様に記述できる。

[手続き CDB-Search]
{可到達行列 B^* から弱連結成分を求める手続き}
1. for i=1 to n do
2. for j=1 to i do
3. if $b^*_{ij}=1$
5. then (要素 i はすでにある要素 j を含む弱連結成分に含まれる。)
6. else (要素 i は新たに弱連結部分成分を構成する)

5. アルゴリズムの構成

3. と 4. の結果を用いた 10 のクラスタリングアルゴリズムをインプリメントした。それぞれ, AD_A, AD_B, CL_A, CL_B, PAD_A, PAD_B, PCL_A, PCL_B, CL_FA (CL_FB) と呼ぶことにする。(図1)

以下でアルゴリズム AD_A～CL_FB について述べる。

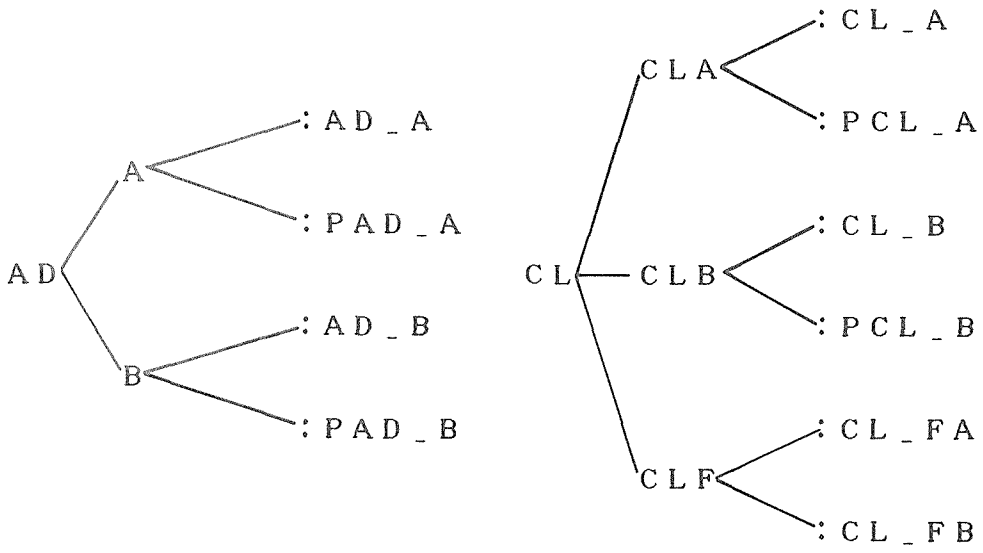

図1 アルゴリズムの分類

5. 1 アルゴリズム

(1) アルゴリズム AD_A と AD_B

アルゴリズム AD_A (AD_B) は弱連結成分の探索に隣接行列A (B) を利用する。

{アルゴリズム AD_A}
1. begin
2. repeat

3．　しきい値tを決定する
4．　行列Fをtにより二値化し A_t を求める。
5．　A_t を用い，再帰的手続きによって，弱連結成分を探索し，しきい値tにおけるクラスターを求める。
6．until すべてのしきい値が終了するまで
7．end

アルゴリズム AD_B は AD_A のステップ5を次のステップで置き換える。

5．1　$B_t=A_t+(A_t)^T$
5．2　B_t を用い，再帰的手続きによって，弱連結成分を探索し，しきい値tにおけるクラスタを求める。

---

(2)　アルゴリズム CL_A と CL_B

CL_A (CL_B) は，A(B) の推移的閉包を用いて弱連結成分を探索する。

{アルゴリズム CL_A}

1．begin
2．　repeat
3．　しきい値tを決定する
4．　行列Fをtにより二値化し A_t を求める。
5．　手続き B-Warshall により，行列 A_t の可到達行列 A^*_t を求める。
6．　A^*_t を用い，手続き CD-Search によって，弱連結成分を探索し，しきい値tにおけるクラスターを求める。
7．　until すべてのしきい値が終了するまで
8．end

アルゴリズム CL_B は CL_A のステップ5と6を次のステップで置き換える。

5．1　$B_t=A_t+(A_t)^T$
5．2　手続き SB-Warshall により，行列 B_t の可到達行列 B^*_t を求める。
6．　B^*_t を用い，手続き CD-Search によって，弱連結成分を探索し，しきい値tにおけるクラスターを求める。

以下のアルゴリズム PAD_A～PCL_B はしきい値を常に $s<t$ となるように選ぶ。

(3)　アルゴリズム PAD_A と PAD_B

アルゴリズム PAD_A (PAD_B) は隣接行列 A(B) を用いて弱連結成分を探索する。一番最初のしきい値についてだけは，AD_A (AD_B) を適用する。

{アルゴリズム PAD_A}

1．begin

2. しきい値sを決定し，アルゴリズムAD-Aを適用し，しきい値sにおけるクラスターを求める。
3. repeat
4. s<tとなるように，しきい値tを選ぶ。
5. tにより行列Fを二値化し，二値行列A_tを得る。
6. A_tにおいて新たに0から1となったすべての成分a_{ij}のうち，要素iと要素jが異なるクラスターに属しているものについて，それぞれの属しているクラスターをまとめて一つのクラスターとする。
7. s=tとセットする。
8. until すべてのしきい値が終了するまで
9. end

アルゴリズムPAD_BはPAD_Aのステップ2と6を次のステップで置き換える。

---

2. しきい値sを決定し，アルゴリズムAD_Bを適用し，しきい値sにおけるクラスターを求める。
6.1 $B_t=A_t+(A_t)^T$
6.2 B_tにおいて新たに0から1となったすべての成分b_{ij}のうち，要素iと要素jが異なるクラスターに属しているものについて，それぞれの属しているクラスターをまとめて一つのクラスターとする。

---

(4) アルゴリズムPCL_AとPCL_B

アルゴリズムPCL_A (PCL_B)はしきい値sにおける可到達行列A^*_s(B^*_s)に対して手続きPA(PSB)-Warshallにより，しきい値tにおける可到達行列A_t^*(B^*_s)を求め，A_t^*(B^*_s)を使ってクラスターを生成する。一番最初のしきい値についてだけはCL_A (CL_B)によりクラスターを求める。

{アルゴリズムPCL_A}

1. begin
2. しきい値sを決定し，CL_Aを適用して，しきい値sにおけるクラスターを求める。
3. repeat
4. t>sとなるように，しきい値tを選ぶ。
5. tにより行列Fを二値化しA_tを求める。
6. $A_t=A_s^*+A_t$
7. A_tの推移的閉包A_t^*を手続きPA-Warshallにより計算する。
8. A_t^*を用い，手続きCD-Warshallによって，弱連結成分を探索し，しきい値tにおけるクラスターを求める。
9. s=tとセットする。
10. until すべてのしきい値が終了するまで
11. end

アルゴリズム PCL_B は PCL_A のステップ2と6，7，8を以下のステップで置き換える。

2．しきい値 s を決定し，CL_B を適用して，しきい値 s におけるクラスターを求める。
6．　$B_t=(B_s^*)+A_t+(A_t)^T$
7．　B_t の推移的閉包 B_t^* を手続き PSB-Warshall により計算する。
8．　B_t^* を用い，手続き CD-Warshall によって，弱連結成分を探索し，しきい値 t におけるクラスターを求める。

(5)　アルゴリズム CL_FA と CL_FB

CL_FA (CL_FB) では最初に類似度行列Fに対して，その閉包行列 F^* を求め，次に F^* をしきい t 値で二値化する。

{アルゴリズム CL_FA}
1．begin
2．　repeat
3．　　一般化 Warshall アルゴリズムにより相互連関行列Fの閉包行列 F^* を求める
4．　　しきい値 t を決定する
5．　　行列 F^* を t により二値化し A^*_t を求める。
6．　　A^*_t を用い，手続き CDA-Search によって，弱連結成分を探索し，しきい値 t におけるラスターを求める。
7．　until すべてのしきい値が終了するまで
8．end

アルゴリズム CL_FB は CL_FA のステップ6を以下のステップで置き換える。

6．1　$B^*_t=A^*_t+(A^*_t)^T$
6．2　B^*_t を用い，手続き CDB-Search によって，弱連結成分を探索し，しきい値 t におけるクラスターを求める。

上のステップ6．1で，$B^*_t=A^*_t+(A^*_t)^T$ とおいて良いことは次の様にして示すことができる。

$X=A^*_t+(A^*_t)^T$ とおく。可到達行列の条件，$X+1=X$，$X^2=X$ を示す。$X+1=1$ は F の定義から明らか。また，x_{ij}，x_{ji} は2値変数であるから $(x_{ij}+x_{ji})^2=x_{ij}+x_{ji}$.

5．2　理論的計算量

アルゴリズム AD_A～CL_FB について，アルゴリズム全体の計算量，閉包をとる計算量，弱連結成分探索の計算量についてまとめると，表1のようになる。閉包を計算してから弱連結成分の探索を行なうアルゴリズムでは，閉包を

表1　各アルゴリズムの計算量

アルゴリズム	全体	弱連結探索	閉包
AD_A, AD_B	$O(n^2)$	$O(n^2)$	−
PAD_A, PAD_B	$O(n^2)$	$O(n^2)$	−
CL_A	$O(n^3)$	$O(n^2)$	$O(n^3)$
CL_B	$O(n^2)$	$O(n^2)$	$O(n^2)$
PCL_A	$O(n^3)$	$O(n^2)$	$O(n^3)$
PCL_B	$O(n^2)$	$O(n^2)$	$O(n^2)$
CL_F_A	$O(n^3)$	$O(n^2)$	$O(n^3)$
CL_F_B	$O(n^3)$	$O(n^2)$	$O(n^3)$

求める部分の計算量が，アルゴリズム全体の計算量を決定していることがわかる。ただし，部分修正のアルゴリズムである PCL-A，PCL-B は，最初に O（n^3）の計算が必要であるが，その後は O（n^2）で済む。

6．アルゴリズムの比較実験

4．で述べた5種類のアルゴリズムについてその処理時間を比較した。

6．1 実験方法

次の示すような手順で，クラスタリングを行うときに必要とされる処理時間を測定した。実験環境は，ハードウェアとしては SORD 社の M 685，OS は UNOS，アルゴリズム記述言語は C である。

［実験方法］

1．次の要領で生成した n 次正方行列 F を類似度行列とする。

a）行 i，列 j の値 i，j を 1～n の乱数を発生させて決定する。

b）0～1000 の非負の整数を乱数で発生させて，f_{ij} の値とする。ただし，ここでの i，j の値は，a）で決定されているものである。

c）F の全要素数（n^2）に対して，正の要素数の割合がある値に達したならば，F の生成を終了させる。この実験では，正の要素数が全要素数の 40％を超えたところで終了とすることにした。

2．クラスタリングは次の要領で実行する。

a）20 の異なるしきい値 t について実行する。

b）$0 \leqq f_{ij} \leqq 1000$（i，j＝1，…，n）であり，あまり小さい値をしきい値に選んでも無意味なので，しきい値 t は，$420 \leqq t \leqq 900$ の範囲で 20 きざみで選ぶことにする。

3．ステップ1，2を数回繰り返し，その総計の時間を計測する。繰り返しの回数は，原則として 10 回としたが，一部のアルゴリズムにおいてはかなり長い処理時間を必要とするため，それらについては，繰り返しの回数を5回とすることにした。

4．ステップ1～3を F の次数 n を 20～200 まで 20 おきに順次変化させて，それぞれの n について実験をおこなった。

5．ステップ3で求まった処理時間から，ステップ1で必要とされる処理時間を差し引き，真にクラスタリングにかかった時間を求めて，それからクラスタリング一回当たり（20 個のしきい値に対する）平均処理時間を求める。

6．2 実験結果の考察

実験の結果を，横軸に要素数をとり，縦軸に処理時間をとったグラフに表わしたのが図2である。

図2から，処理時間としては，AD-B を用いたアルゴリズムが最も速く，次いで PAD-B，PAD-A，PCL-B が速い。但し，AD-B と PAD-B はほとんど差がない。AD-B が PAD-B より速いのは次の理由によると考えられる。AD-B では新しいしきい値 t に対して，2値化と対称化を行い B_t を生成し，その後，弱連結成分の探索を行う。これに対して PAD-B では，B_t を生成する他に，前のしきい値 s での対称2値行列 B_s との比較を行い，新たに1がたった要素を見つけなければならない。この演算は行列の要素数に等しく O（n^2）である。結局，AD-B が極めて速いため $n \leqq$

200 では PAD_B のオーバーヘッドが大きくて AD_B の方が速い結果となったと考えられる。

要素数が 100 のときと要素数が 200 のときの処理時間に注目すると，要素数が 2 倍になったのに対して，処理時間は，CL_A (B)，CL_F_A (B)，では約 8 倍，PCL_A (B)，AD_A (B) では約 4 倍，PAD_A (B) では約 2 倍となっている。このことは，実際に行なわれる計算量が，表 1 で示される理論的な計算量と，ほぼ等しいものであることを示している。全体として，

(1) 閉包行列を利用するアルゴリズムは前処理に占める時間の割合が大きく，全体の計算時間が長くなる。しかし，プログラムは再帰呼び出しの機能がない言語でもきわめて容易である，

(2) 隣接行列を利用するアルゴリズムは前処理の必要がなく計算時間は速い。しかし，再帰呼び出しの機能がない言語ではプログラムが複雑になる。再帰呼び出しの機能がある言語が利用できるときはこの機能を利用することによりプログラムも簡潔に記述できる，

と言える。

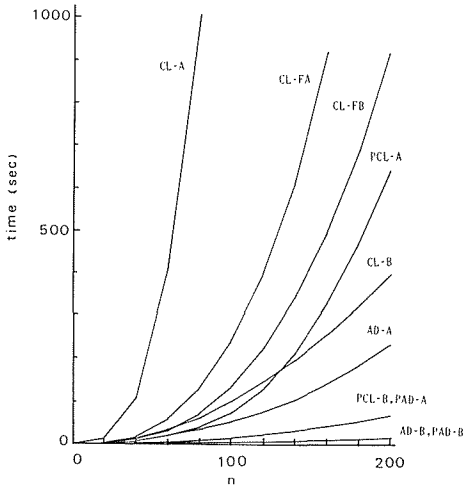

図 2．　アルゴリズムの処理時間

以上をまとめると，再帰呼び出しの機能がある言語を使用するときは AD_B が適当と思われる。また，再帰呼び出しの機能がない言語を利用するときは PCL_B が適当である。

7．む　す　び

弱連結成分抽出によるグラフ理論的クラスター分析のアルゴリズムについて検討した。理論解析および実験結果から，しきい値を単調に減少させながら隣接行列を利用しクラスター分析を行なうアルゴリズム AD_B，あるいは可到達行列を利用するアルゴリズム PCL_B などが適していると思われる。これらのアルゴリズムはインプリメントも容易であり，16 ビットパソコンでも実用規模の問題を解くことができ，応用ソフトのアルゴリズムとして利用価値が高いと思われる。

参 考 文 献

1) J. A. Hartigan : "Clustering Algorithms", John Wiley & Sons, Inc. (1975).
2) tsiNKy : "ちょっとめずらしい算法の話し", ビット，13，pp. 83_86，(1981).
3) J. P. Boyd : "Asymmetric Clusters of Internal Migration Regions of France", IEEE Trans. Syst. Man, Cybern., SMC_10, 2, pp. 101_105(1980).
4) 大内，加地："超計量行列に対する一考察"，信学論 70_A，5 号 (昭 62)
5) S. Warshall : "A theorem on Boolean matrices", J. of ACM, 9, pp. 11_12(1962).
6) M. Yoeli : "A note on a generalization of Boolean matrix theory", Amer. Math. Monthly, 68, pp. 552_557, (1961).
7) P. Robert and J. Ferland : "Generalisation de l'algorithme de Warsahll", R. 1. R. 0., 2, pp. 71_85, (1968).
8) A. T. Berztiss : "Data Structures, Theory and Practice", second edition, pp. 270, Academic Press (1975)
9) A. V. Aho, J. E. Hopcroft & J. D. Ullman : "The design and Analysis of Computer Algorithms", Addison Wesley (1974).