

## プログラムのページ

担当 吉 沢 正

### 7103 ネットワークのグループ分類について

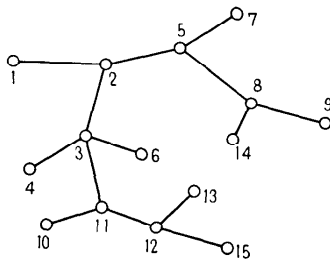
村山康宏 (釧路工業高等専門学校)

#### 1. 序 論

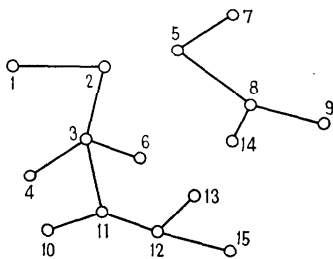
ひとつのネットワークにおいて、枝が切り落された場合に、そのネットワークは一般に2個以上のサブネットワークに分割される。ネットワークが、たとえば電力系統網である場合を考えると、各ノードがどのサブネットワークに属しているかを自動的に判別する必要がある。これは系統分離が発生して、各サブネットワーク内での変動量（たとえば、周波数変動、負荷変動など）を計算する場合がそうである。本プログラムはノードの分類を行なうために開発されたものである。

#### 2. アルゴリズム

いま、第1図のネットワークで代表させて考える。ノード2とノード5との間の枝が脱落した場合を考え



第1図 ネットワーク



第2図 サブネットワーク

ると第2図のようになる。

ここで

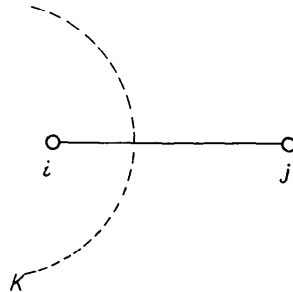
$S_i$ :  $i$ ノードの属するグループ.

$N$ : ノードの個数.

$E_{i,j}$ :  $i$ ノードと $j$ ノード間の接続行列で、オンならば1, オフならば0である.

$K$ : グループ番号.

$l$ : ノードを表わす補助変数.



第3図

- Step 1. 全ノードのグループを零にセットする.  
 $S_i = 0$  ( $i = 1 \sim N$ )
- Step 2. グループ番号 $K$ , 補助変数 $l$ をともに1にセットする.
- Step 3.  $l < N$  ならば終了,  $l \leq N$  ならば Step 4へ.
- Step 4. 検索するノード $i$ を $l$ にセットする.  
(最初の段階では当然  $i = 1$  である.)
- Step 5.  $S_i = K$  ならば Step 19へ,  $S_i \neq K$  ならば Step 6へ.
- Step 6.  $S_i = 0$  (ノードの未登録) ならば Step 7へ.  $S_i \neq 0$  ならば Step 8へ.
- Step 7.  $S_i$ を $K$ にセットする.  $i$ ノードはグループ $K$ に登録された.
- Step 8.  $K = 1$  ならば Step 7へ,  $K \neq 1$  ならば $l$ を1だけ増して Step 3へ.
- Step 9. Step 7までの段階で $i$ ノードの属するグループが決まっているので, Step 9以

## プログラム

.....	FACOM 230-60	FORTRAN D	-700*10-	000*+03	.....	COMPILATION	70.12.24	PAGE	3
* SOURCE STATEMENT *									
1		SUBROUTINE	PATH					P	1
2		DIMENSION	E(50,50),C(50,50),SN(50),I(50),J(50)					P	2
3		COMMON	E,C,SN,I,J,NAP,IL,K,L,JL,MAX,KMAX,KI,KJ,JJ					P	3
4		DO	309 IL=1,NAP					P	4
5		309	SN(IL)=0					P	5
6		K=1						P	6
7		L=1						P	7
8		6666	IF(L=NAP-1) 499,401,401					P	8
9		499	IL=K					P	9
10		402	IF(SN(IL)=K) 404,403,404					P	10
11		404	IF(SN(IL)) 405,408,405					P	11
12		408	SN(IL)=K					P	12
13		DO	409 JL=1,NAP					P	13
14		IF	(E(IL,JL)) 410,409,410					P	14
15		410	IF(SN(JL)=SN(IL)) 411,409,411					P	15
16		411	IF(CIL=JL) 412,413,412					P	16
17		412	IF(SN(IL)=JL) 413,414,413					P	17
18		413	SN(IL)=SN(JL)					P	18
19		416	IL=JL					P	19
20		GO	TO 402					P	20
21		413	IF(SN(JL)=K) 414,409,414					P	21
22		414	SN(IL)=SN(JL)					P	22
23		GO	TO 416					P	23
24		409	CONTINUE					P	24
25		GO	TO 417					P	25
26		405	IF(K=1) 407,408,407					P	26
27		407	L=L+1					P	27
28		GO	TO 6666					P	28
29		403	DO 418 JL=1,NAP					P	29
30		IF	(E(IL,JL)) 419,418,419					P	30
31		419	IF(SN(JL)=SN(IL)) 420,418,420					P	31
32		420	IF(SN(JL)) 421,421,422					P	32
33		421	SN(JL)=SN(IL)					P	33
34		IL=JL						P	34
35		GO	TO 403					P	35
36		422	IF(SN(JL)=SN(IL)) 424,418,423					P	36
37		424	SN(IL)=SN(JL)					P	37
38		GO	TO 403					P	38
39		423	SN(JL)=SN(IL)					P	39
40		IL=JL						P	40
41		GO	TO 403					P	41
42		418	CONTINUE					P	42
43		GO	TO 417					P	43
44		417	L=L+1					P	44
45		MAX=SN(1)						P	45
46		DO	710 IO=1,NAP					P	46
47		IF	(MAX=SN(IO)) 711,710,710					P	47
48		711	MAX=SN(IO)					P	48
49		710	CONTINUE					P	49
50		K=K+1						P	50
51		GO	TO 6666					P	51
52		401	KMAX=SN(1)					P	52
53		DO	810 IO=1,NAP					P	53
54		IF	(KMAX=SN(IO)) 811,810,810					P	54
55		811	KMAX=SN(IO)					P	55
* SOURCE STATEMENT (PATH) *									
56		811	KMAX=SN(IO)					P	55
57		810	CONTINUE					P	56
58		DO	40001 KI=1,KMAX-1					P	57
59		DO	40002 KJ=KI+1,KMAX					P	58
60		DO	40003 IL=1,NAP					P	59
61		DO	40004 JL=1,NAP					P	60
62		IF	(SN(IL),NE,KJ) GO TO 40003					P	61
63		IF	(SN(JL),NE,KI) GO TO 40004					P	62
64		IF	(E(IL,JL),EQ,IO) GO TO 40004					P	63
65		DO	40005 JJ=1,NAP					P	64
66		IF	(SN(JJ),NE,KJ) GO TO 40005					P	65
67		SN(IJ)=SN(IL)						P	66
68		40005	CONTINUE					P	67
69		40004	CONTINUE					P	68
70		40003	CONTINUE					P	69
71		40002	CONTINUE					P	70
72		40001	CONTINUE					P	71
73		WRITE	(6,3458) (SN(IO),IO=1,NAP)					P	72
74		3458	FORMAT(1H,5X,2015/)					P	73
75		RETURN						P	74
76		END						P	75







後では  $i$  ノードと接続している  $j$  ノードを捜すためのループのセットを行なう。

- Step 10. 接続行列  $E_{i,j}=0$  の判別を行ない,  $i$  ノードと  $j$  ノードとの接続の有無を調べる.  $E_{i,j}=0$  ならば  $j$  を 1 だけ増して Step 9 へ,  $E_{i,j} \neq 0$  ならば Step 11 へ.
- Step 11.  $S_j=S_i$  ならば,  $i$  ノードと  $j$  ノードとは同じグループに属しているので  $j$  を 1 だけ増して Step 9 へ.  $S_j \neq S_i$  ならば, 一応  $i$  ノードと  $j$  ノードとは異なるグループに属しているので Step 12 へ.
- Step 12.  $i < j$  ならば Step 13 へ,  $i > j$  ならば Step 16 へ.
- Step 13.  $S_i=1$  ならば Step 17 へ,  $S_i \neq 1$  ならば Step 14 へ.
- Step 14. この時点では  $S_i \neq 1$  であるから,  $S_i$  を  $S_j$  にセットする. その理由は  $S_j$  は  $S_i$  よりも先に決定しているからである.
- Step 15. この時点で  $i$  ノード  $j$  ノードとのグループが決定したので, 新たな  $i$  ノードを  $j$  ノードにセットして Step 5 へ.
- Step 16.  $S_i=K$  ならば  $j$  を 1 だけ増して Step 9 へ,  $S_i \neq K$  ならば Step 17 へ.
- Step 17.  $S_j$  を  $S_i$  にセットして Step 15 へ.
- Step 18.  $l$  を 1 だけ増し ( $i$  ノードと接続している  $j$  ノードをたどり終えたので), グループ番号をこの時点のグループ番号の最大値に 1 だけ加えて Step 3 へ.
- Step 19.  $i$  ノードと接続している  $j$  ノードを捜すためのループのセット.
- Step 20.  $E_{i,j} \neq 0$  ならば  $S_j$  のセットを行ない,  $E_{i,j}=0$  ならば  $j$  を 1 だけ増して Step 19 へ.
- Step 21. 次の  $i$  ノードが決ったならば Step 19 へ.
- Step 22.  $j < N$  ならば,  $l$  を 1 だけ増し,  $K$  をこの時点の最大値より 1 だけ増して Step 3 へ.

以上のアルゴリズムでノードのグループ分類を行なう. 本プログラムによれば, ノードの個数と接続行列

とを入力データとして与えるだけで, 各枝の切り落としによるグループ分類が可能であり, プログラムのステートメントの数も比較的少なくなるとすむ. なお, 演算時間は本例題によると数十ミリ秒程度である. ノードの個数が同一であっても, ノード間の接続行列の状態によって計算時間に差が生ずる.

### 3. プログラムおよびモデルネットワークに関する結果

おもな変数

$C(,)$ : 接続行列をストアするためのテーブル.

$I( )$ : ノード.

$J( )$ : ノード.

$E(,)$ : ノード間の接続行列.

$K$ : グループ番号.

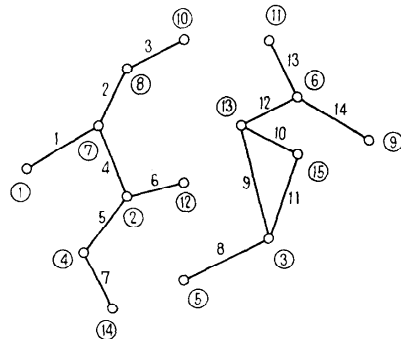
$N$ : 枝の数.

$NAP$ : ノードの数.

$L$ : ノードを表わす補助変数.

$L1$ : 脱落枝の番号.

$L2$ : 脱落枝の番号.



第4図 モデルネットワーク

本プログラムは, 2枝同時脱落までについて取り扱っている. ただし, 入力データは第4図のとおりである. なおメインプログラムは入力データの読み込みおよび枝脱落の模擬の部分であるから省略して, グループ決定のサブプログラムのみを記述する.

(昭和46年2月24日受付)