

## 8.2 ALGORISME D'ASSIGNACIO

### 8.2.1. ESTRUCTURA GENERAL.

L'algorisme d'assignació s'ha programat en llenguatge FORTRAN - IV. La versió actual està preparada per treballar en un sistema FACOM 230-25. Consta de 15 mòduls, a part del programa principal, i ocupa unes 70 K de memòria central; per a una ciutat amb 44 zones. Admet la possibilitat de funcionar tot ell com una subrutina del programa de generació. Comprèn l'execució íntegra de l'algorisme d'assignació, a més de la preparació de dades d'entrada i de la impressió de resultats de sortida.

### 8.2.2. RELACIO D'ARXIUS.

Un programa llegeix 4 arxius, en aquest ordre:

- a) matriu de demandes (en nombre de viatgers).
- b) " de cost a peu (en min.).
- c) " " " en bus (en min.).
- d) xarxa, com a conjunt de línies (línia, com a enfilall de nusos).

La xarxa es llegeix línia per línia, a raó d'un registre lògic per línia. Aquest mode de lectura és degut a dos motius.

- controlar la correcta definició de les línies.
- tractar i preparar les dades de cada línia per separat.

### 8.2.3. SORTIDES O INFORMACIO FORNIDA PEL PROGRAMA.

Les sortides són sempre per impressora.

De fet, l'output principal, és a dir, el nombre de busos assignats a cada línia, és un vector de, com a màxim, 40 elements. Seria, doncs, molt poc rendible d'haver de crear un arxiu en disc o cinta per a emmagatzemar-lo. L'única sortida en principi interessant, és a dir, per targeta, és impossible ja que el sistema FACOM no té perforadora.

Totes les altres sortides, com tot seguit es veurà tenen un interès purament il·lustratiu, que no justifica en principi un emmagatzement per a ser tractades ulteriorment.

### 8.2.3.1. Paràmetres del procés.

Es un llistat imprimit al començament del programa, que assabenta del valor d'alguns paràmetres i variables rellevants per tal de caracteritzar aquella execució en concret.

A la figura 8.2.1. s'hi adjunta la llista de variables impreses.

Tots els llistats, corresponen a una prova de la xarxa X44.

NOM	NOM O EQUIVALENT EN L'ALGORISME	NOM EN EL PROGRAMA	UNITAT EN QUE S'EXPRESSA
Nombre de zones	n	NN	-
Nombre de línies	m	MRE	-
Nombre de busos	A	AUT	-
Longitud de la xarxa	$\sum_{i=1}^n D_i$	SX	Km.
Velocitat a peu	-	V	m/min.
Velocitat en bus	v	YBUS	m/min.
Penalització de l'espera	W	PE	-
Equivalent en temps de la tarifa	E	E	min.
Sensibilitat a la dife_rència de costos	$\beta$	BETA	min.
Diferència amb el cost mínim.(indicador de sensib.)	h	HAC	min.
Població activa total	$\sum_i \sum_j d_{ij}$	IW	viatges.
Població potencial absorbible (sota hipòtesi de busos infinits)	-	IP	viatges.
Població efectivament absorbida al començ del programa	$\sum_l d_l$	ID	viatges.
Cost inicial unitari	$C/\sum \sum d_{ij}$	W	min./viatge.
Cost inicial mínim (sota les hipòtesis d'abans)	-	W2	min./viatge.
Periodicitat de les descrip. de la xarxa	-	NREV	canvis.

FIG. 8.2.1

8.2.3.2. Classificació dels fluxos segons el nombre de línies.

S'imprimeix al començament i al final de programa, per tal de comparar les variacions que hi ha hagut durant l'execució.

Consisteix en un llistat de 5 columnes, amb la següent informació:

- 1ª col) Nombre de línies (de 0 a 8 per flux) l.
- 2ª col) " " parells de nusos connectats per 1 línies.
- 3ª col) Demanda NO absorbida, corresponent a cada grup de parells de nusos.
- 4ª col) Demanda SI absorbida.
- 5ª col) Demanda total.

Al capdavant, s'imprimeixen les sumes de cada columna.

RELACIO DE LES UNIONS ENTRE PARELLS DE NUSOS

\*\*\*\*\*

NO, LINES	NO, PARELLS DE NUSOS	NO ABSORBIDA	DEMANDES ABSORBIDA	TOTAL
0	181	135292	0	135292
1	354	7310	119454	126764
2	275	16076	160888	176964
3	105	10762	103161	113944
4	45	16808	70255	87064
5	17	8744	35121	43866
6	10	8976	18377	27354
7	2	7588	5067	12656
8	1	2696	1059	3956
TOTALS	990	214456	513405	727861

PARAMETRES DEL PROCES

\*\*\*\*\*

NOMBRE DE ZONES	44
NOMBRE DE LINES	26
NOMBRE DE BUSOS	500
LONGITUD DE XARXA	430.239 KM.
VELOCITAT A PEU	66.667 M./MIN.
VELOCITAT DE BUS	200.000 M./MIN.
PENALITZACIO ESPERA	1:
EQUIVALENT DE LA TARIFA	10: MIN.
SENSIBILITAT A LA DIF. DE COSTOS	2.000
DIFERENCIA AMB EL COST MINIM	0.50 MIN.
POBLACIO ACTIVA TOTAL	727854
POBLACIO POTENCIAL ABSORBIBLE	543312
POBLACIO ABSORBIDA PER LA XARXA	513968
COST INICIAL	26.76015 MIN.
COST MINIM	24.72220 MIN.
PERIODICITAT DE LES DESCRIPCIONS	90 CANVIS

### 8.2.3.3. Estat de la xarxa. (Llistat reduït).

S'imprimeix al començament després de les assignacions inicials, cada NREV canvis (vegi's la taula 8.2.1.) operats per l'algorisme, i al final del procés.

Per a cada línia imprimeix:

- el seu número identificatiu.
- el nombre de nusos (no la llista) que la constitueixen.
- la longitud en Km.
- la demanda absorbida.
- l'interval en min.
- el nombre de busos assignats.

ESTAT DE LA XARXA DESPRES DE LA REASSIGNACIU 3  
 \*\*\*\*\*

LINIA	NUSOS	LONGITUD (K.M.)	DEMANDA	INTERVAL (MIN.)	BUSOS	LINIA
1	10	12.56	18137	6.28	20	1
2	15	20.72	31545	6.10	34	2
3	11	17.00	14135	8.10	21	3
4	11	16.77	6484	11.98	14	4
5	12	16.89	27676	5.82	29	5
6	12	17.52	12808	8.76	20	6
7	10	12.67	10724	7.92	16	7
8	14	20.94	17558	8.05	26	8
9	10	14.24	7907	10.17	14	9
10	11	20.73	14474	9.01	23	10
11	11	14.45	10936	8.50	17	11
12	11	16.03	13866	6.01	20	12
13	11	15.45	12840	8.13	19	13
14	13	18.61	22466	6.89	27	14
15	13	18.38	22549	6.81	27	15
16	16	19.48	18872	7.49	26	16
17	15	21.22	22283	7.32	29	17
18	9	12.84	11083	8.03	16	18
19	11	17.62	11237	9.27	19	19
20	11	16.79	8396	10.49	16	20
21	12	16.36	11035	9.09	18	21
22	11	15.21	18012	6.91	22	22
23	12	18.64	11618	9.32	20	23
24	5	6.91	4009	9.87	7	24

#### 8.2.3.4. Estat de la xarxa. (Llistat complet).

Es alternatiu de l'anterior. Segons el valor d'una variable de control (ILLIS), s'imprimeix el reduït o el complet, en els mateixos instants d'execució.

Forneix, a més a més, per a cada línia.

- la llista de nusos.
- la càrrega a cada tram (entre nus i nus).

Comparat amb el llistat anàleg del programa de generació, pot veure's que ací s'ha omès la longitud i la demanda absorbida per la xarxa. Allà, aquestes dades estaven justificades perquè la revisió d'una línia ja hi implicava un canvi; ara, però, es contemplen totes les línies globalment, i n'hi ha prou de donat aquestes xifres una vegada.

Tampoc no es fa cap referència al temps de recorregut entre nusos car aquí, en no alterar-ne el traçat, aquells romanen constants, la qual cosa no s'esdevenia a l'algorisme de generació.





LINIA \* 9\*

DEM, 22938 ORD, 1 2 3 4 5 6 7 8 9 10 11 12 13  
 BUSOS 22 CAR, 2231 3776 4581 5346 6674 9474 9037 7308 4987 3158 1945 1240  
 INT, 8,03 NUS 6 12 14 18 21 23 27 31 34 35 37 39  
 LONG, 17,67

LINIA \*10\*

DEM, 30034 ORD, 1 2 3 4 5 6 7 8 9 10 11 12 13  
 BUSOS 33 CAR, 10268 7953 6981 6299 5927 4334 4158 4979 5913 7575 7834 7493  
 INT, 5,88 NUS 26 20 19 24 29 32 37 35 34 31 28 27 26  
 LONG, 19,39

LINIA \*11\*

DEM, 28230 ORD, 1 2 3 4 5 6 7 8 9 10 11 12  
 BUSOS 24 CAR, 2232 4112 5869 5955 6302 7947 11082 12513 10162 7876 3732  
 INT, 7,11 NUS 44 42 40 39 37 33 25 20 22 23 21 18  
 LONG, 17,05

LINIA \*12\*

DEM, 11813 ORD, 1 2 3 4 5 6 7 8  
 BUSOS 15 CAR, 3554 4596 5368 7023 6013 4003 2553  
 INT, 7,51 NUS 40 39 37 33 30 26 27 28  
 LONG, 11,26

LINIA \*13\*

DEM, 32605 ORD, 1 2 3 4 5 6 7 8 9 10 11 12  
 BUSOS 30 CAR, 3176 4740 6517 7371 7289 8671 12354 11414 8219 6204 3485  
 INT, 5,31 NUS 1 6 10 13 17 22 26 25 30 31 34  
 LONG, 15,92

LINIA \*14\*

DEM, 19415 ORD, 1 2 3 4 5 6 7 8 9 10 11 12 13 14  
 BUSOS 25 CAR, 2343 2597 2982 4062 4240 5294 5486 5185 4855 3910 3629 3212 1802  
 INT, 8,75 NUS 4 8 12 14 18 22 26 30 35 38 41 43 42 44  
 LONG, 21,88

LINIA \*15\*

DEM, 8251 ORD, 1 2 3 4 5 6 7 8 9 10  
 BUSOS 10 CAR, 1405 3194 2832 2352 2411 1877 1951 1733 1069  
 INT, 13,75 NUS 18 21 28 31 34 38 41 43 42 44  
 LONG, 13,75

LINIA \*16\*

DEM, 13252 ORD, 1 2 3 4 5 6 7 8 9 10  
 BUSOS 18 CAR, 1642 2173 2699 2828 4381 4834 6264 5438 3044  
 INT, 7,9 NUS 1 1642 2 2173 6 2699 9 2828 16 4381 17 4834 22 6264 23 5438 28 3044  
 LONG, 14,23



LINIA \*24\*

DEM.	22920	ORD.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
BUSOS	24	CAR.	1696	1879	2625	3928	4453	6463	8152	8458	7560	7099	5209	4465	3513	1954	
INT.	8.77	NUS	4	8	12	13	17	22	23	27	31	34	38	41	43	42	44
LONG.		21.05															

LINIA \*25\*

DEM.	6417	ORD.	1	2	3	4	5	6	7	8
BUSOS	9	CAR.	398	1706	2394	2640	2481	2600	1349	
INT.	11.63	NUS	5	6	10	12	14	18	23	27
LONG.		10.46								

LINIA \*27\*

DEM.	19748	ORD.	1	2	3	4	5	6	7	8	9	10	11
BUSOS	25	CAR.	1563	4115	6125	6756	7388	7516	8128	6938	6709	2795	
INT.	6.6	NUS	19	24	29	33	30	31	34	38	41	40	39
LONG.		16.68											

#### 8.2.3.5. Assignacions sota hipòtesi de línia única.

La primera part de l'algorisme s'ocupa d'assignar busos sota hipòtesi de línia única, tal com ja s'ha explicat abans.

En aquest procés, se segueix un mètode iteratiu; al final de cada iteració s'imprimeix:

- per cada línia:
  - . el seu número identificatiu.
  - . el nombre de busos anterior.
  - . el nombre de busos actual.
  - . la diferència entre aquest dos valors.
- el nombre total de busos que s'han canviat de línia.
- la demanda absorbida per la xarxa.
- el cost unitari després de l'assignació.
- el guany percentual en cost obtingut, entès com a tant per cent de la diferència entre el cost inicial i el mínim.
- el guany percentual acumulat.

REASSIGNACIO 1  
\*\*\*\*\*

LINIA	BUSOS ANTERIORS	BUSOS ACTUALS	-SALDO
1	22	25	3
2	6	3	-3
3	15	19	4
4	20	26	6
5	26	24	-2
6	15	18	3
7	14	14	0
8	17	16	-1
9	21	23	2
10	23	25	2
11	20	23	3
12	13	13	0
13	18	22	4
14	25	24	-1
15	16	15	-1
16	17	16	-1
17	18	17	-1
18	16	14	-2
19	20	23	3
20	21	21	0
21	26	22	-4
22	26	19	-7
23	20	18	-2
24	24	23	-1
25	12	11	-1
26	8	7	-1
27	19	18	-1
28	2	1	-1

NOMBRE DE CANVIS 30 BUSOS  
DEMANDA XARXA 515406  
COST MITJA 26.72874 MIN.  
GUANY PERCENTUAL 1.54089  
GUANY ACUMULAT 1.54089

#### 8.2.3.6. Informació sobre els canvis.

La segona part de l'algorisme s'ocupa de canviar autobusos d'una línia a una altra, d'un en un.

A cada canvi, el programa imprimeix.

- per a les línies augmentada i disminuïda:

- . el seu número identificatiu.
- . el nombre actual de busos.
- . l'interval actual en min.
- . l'interval anterior (amb un autobús de menys o de més).

	BUSOS		INTERVAL		BUSOS	INTERVAL		COST MITJA XARXA	DEMANDA	GUANYS		
	ACTUAL	ANTERIOR	ACTUAL	ANTERIOR		ACTUAL	ANTERIOR			PERCENT.	ACUMULAT	
4	*22*	22	10,03	(10,51)	*28*	0	*****	(16,35)	26,69722	515667	0,20590	3,08777
5	*17*	20	7,71	(8,11)	*26*	5	13,45	(11,21)	26,69470	515711	0,12354	3,21131
6	*10*	28	6,93	(7,18)	*26*	4	16,81	(13,45)	26,69229	515749	0,11830	3,32961
7	*5*	26	8,71	(9,06)	*26*	3	22,42	(16,81)	26,68985	515806	0,11980	3,44941
8	*5*	27	8,39	(8,71)	*26*	2	33,63	(22,42)	26,68736	515862	0,12204	3,57146
9	*5*	28	8,09	(8,39)	*26*	1	67,26	(33,63)	26,68481	515916	0,12504	3,69649
10	*13*	24	6,63	(6,92)	*26*	0	*****	(67,26)	26,68208	516014	0,13402	3,83052
11	*5*	29	7,81	(8,09)	*7*	11	10,84	(9,94)	26,67993	516049	0,10527	3,93609
12	*22*	23	9,59	(10,03)	*7*	10	11,92	(10,84)	26,67769	516099	0,11006	4,04615
13	*10*	29	6,69	(6,93)	*7*	9	13,25	(11,92)	26,67542	516116	0,11126	4,15771
14	*5*	30	7,55	(7,81)	*7*	8	14,91	(13,25)	26,67329	516152	0,10407	4,26179
15	*17*	21	7,34	(7,71)	*7*	7	17,04	(14,91)	26,67105	516175	0,11006	4,37185
16	*27*	20	8,34	(8,78)	*7*	6	19,87	(17,04)	26,66876	516183	0,11231	4,48416
17	*4*	30	5,75	(5,95)	*7*	5	23,85	(19,87)	26,66655	516252	0,10827	4,59273
18	*J*	31	7,31	(7,55)	*7*	4	29,81	(23,85)	26,66440	516290	0,10557	4,69830
19	*13*	25	6,37	(6,63)	*7*	3	39,75	(29,81)	26,66216	516364	0,11006	4,80836
20	*10*	30	6,46	(6,69)	*7*	2	59,62	(39,75)	26,65996	516378	0,10722	4,91618

21	* 5*	32	7.08	( 7.31)	* 7*	1	119.25	(59.62)	26,65787	516418	0.10228	5.01875
22	*22*	24	9.19	( 9.59)	* 7*	0	*****	(*****)	26,65564	516465	0.10931	5.12807
23	* 4*	31	5.57	( 5.75)	*21*	18	12.24	(11.59)	26,65387	516509	0.08685	5.21492
24	*13*	26	6.12	( 6.37)	*21*	17	12.96	(12.24)	26,65219	516554	0.08236	5.29728
25	*27*	21	7.94	( 8.34)	*21*	16	13.76	(12.96)	26,65060	516536	0.07787	5.37515
26	*17*	22	7.01	( 7.34)	*21*	15	14.68	(13.76)	26,64912	516533	0.07263	5.44778
27	* 5*	33	6.87	( 7.08)	*21*	14	15.73	(14.68)	26,64790	516548	0.05990	5.50768
28	*10*	31	6.26	( 6.46)	*24*	21	10.02	( 9.57)	26,64673	516495	0.05765	5.56533
29	* 4*	32	5.40	( 5.57)	*15*	13	10.58	( 9.82)	26,64546	516522	0.06214	5.62747
30	*13*	27	5.90	( 6.12)	*18*	12	11.83	(10.92)	26,64441	516540	0.05166	5.67914
31	*22*	25	8.83	( 9.19)	*21*	13	16.94	(15.73)	26,64346	516555	0.04642	5.72556
32	*27*	22	7.58	( 7.94)	*24*	20	10.53	(10.02)	26,64256	516489	0.04418	5.76973
33	*17*	23	6.70	( 7.01)	*15*	12	11.46	(10.58)	26,64162	516468	0.04642	5.81616
34	*23*	19	8.93	( 9.42)	*18*	11	12.90	(11.83)	26,64079	516468	0.04043	5.85659
35	*12*	14	8.04	( 8.66)	*21*	12	16.33	(16.94)	26,64005	516444	0.03669	5.89328







FITXA	NOM DE LA VARIABLE	NOM EN EL PROGRAMA	FORMAT	CONDICIONS
13	Periodicitat de les descripcions de la xarxa (cada NREV canvis, el programa llista les línies)	NREV	(I3, 77x)	
14	Indicador de llistat reduït o complet.	ILLIS	(I3, 77x)	= 0 → reduït. ≥ 0 → complet.
15	Indicador d'assignació inicial donada de busos.	IX	(I3, 77x)	= 0 → NO n'hi ha, i el programa la <u>cal</u> cula. ≥ 0 → SI n'hi ha, i el programa espera llegir-la a la <u>se</u> güent fitxa.
16 (paquet si MRE>25)	Assignació inicial d'autobusos, donada pel programador.	NBUS(NLIN)	(25I3)	. només si IX ≤ 0.

FIG. 8.2.2



### 8.2.5. ESTRUCTURA INTERNA DEL PROGRAMA.

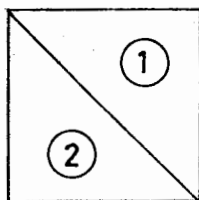
Com ja s'ha dit abans, el programa consta de 15 mòduls: el programa principal i 14 subrutines, més o menys jerarquitzades. El diagrama de la figura 8.2.3., permet de veure com es relacionen entre sí.

A continuació es descriu el paper de cada una.

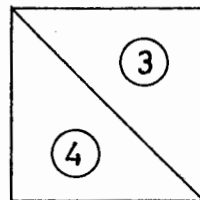
#### 8.2.5.1. Programa principal.

Es el substitut de l'algorisme de creació de la xarxa, quan l'actual algorisme, d'assignació, es vol fer funcionar sol. S'encarrega, doncs, de preparar les dades tal com seria la sortida del primer algorisme:

- a) respecte dels paràmetres:
  - . llegeix NN, AUT, V, Z, PE, E i BETA.
  - . comprova que NN no excedeixi el valor màxim permès.
- b) respecte de les matrius:
  - . llegeix D: demandes.
    - C: distància a peu en m.
    - A: distància en bus en mi.
  - . les triangularitza de la següent manera:



D



C

- ① conté  $d_{ik} + d_{ki}$
- ② conté 0, en previsió d'allotjar-hi el nombre de línies que uneixen (i,k), un cop llegida la xarxa. És a dir,  $[\eta_{ij}]$ , tal com es definia a l'algorisme.
- ③ conté el cost a peu en minuts,  $[p_{ij}]$
- ④ conté el cost en bus en minuts, o temps mínim de recorregut  $[r_{ij}]$

La matriu A desapareix.

- c) respecte de la xarxa.
  - . llegeix el nombre total de línies i comprova que no excedeixi el valor màxim permès.
  - . per a cada línia:
    - comprova que tots els nusos pertanyin a la zona d'estudi, és a dir, que no ultrapassin el valor NN.
    - crea la fila corresponent de la matriu de temps recorregut  $[t_{ih}]$
    - crea la fila corresponent de la matriu de lògica de pertinença  $[g_{ij}]$

Finalment, crida la subrutina FREQ.

\*ID. SEQ.

ISN \* I SOURCE STATEMENT / ERROR MESSAGE

C \*\*\*\*\*

C PROGRAMA INICIAL PER A L'ASSIGNACIO DE BUSOS A LES LINES D'UNA XARXA

C \*\*\*\*\*

```

1 REAL INT
2 INTEGER D, AUT, CIRC
3 LOGICAL
4 COMMON /GRUP 0/ V, Z, TA, AUT, ILLIS
5 /GRUP 1/ NUS(44,40), MPE
6 /GRUP 2/ NL(25,40), T(25,40), NNU(40), CIRC(40), YBUS
7 /GRUP 3/ INT(40,5), NBUS(40), E, PE, BETA, HAC
8 /GRUP 4/ D(44,44), C(44,44), NN, NCO
9 /GRUP 5/ A(44,44)
10 DATA IENT /5/, ISOR /6/
11 DATA NNOH /25/, NNMAX /44/, MREM /40/

```

C ENTRADA DE DADES

C \*\*\*\*\*

C PARAMETRES

```

7 READ(IENT,52) NN, AUT
8 IF(NN.LE.NNMAX) GO TO 1
9 WRITE(ISOR,52) NN, NNMAX
10 STOP
11 READ(IENT,54) V, Z, PE, E, BETA

```

C MARIUS

```

12 READ(IENT,56) ((C(I,K),K=1,NN),I=1,NN)
13 READ(IENT,58) ((C(I,K),K=1,NN),I=1,NN)
14 READ(IENT,59) ((A(I,K),K=1,NN),I=1,NN)

```

C FORMATS D'ENTRADA

```

15 52 FORMAT(13, 77X)
16 54 FORMAT(F10.0, 70X)
17 56 FORMAT(13(15, 1X))
18 58 FORMAT(13F6,10)
19 60 FORMAT(5X, 2513)

```

C ORDENACIO DE MARIUS (TRIANGULARITZACIO)

C \*\*\*\*\*

```

20 NCO = NN - 1
21 YBUS = V * Z
22 HAC = 1. / BETA
23 DO 2 I=1,NCO
24 MO = I + 1
25 DO 2 K=N0,NN
26 D(I,K) = D(I,K) + D(K,I)
27 D(K,I) = 0
28 C(I,K) = C(I,K) / V
29 C(K,I) = A(I,K) / YBUS

```

C CREACIO DE LES MARIUS RELATIVES A LES LINES

C \*\*\*\*\*

\*ID, SEQ.

SOURCE STATEMENT / ERROR MESSAGE

ISN

C INICIALITZACIO

```

30 HEAD(ENT,52) MRE
31 IF (MRE.LT.MREM) GO TO 3
32 WRITE(1509,84) MRE, MREM
33 STOP
34 DO 6 N=1,MRE
35 CIRC(NLIN) = .FALSE.
36 DO 4 N=1,NN
37 NUS(N,NLIN) = .FALSE.
38 DO 6 J=1,25
39 T(J,NLIN) = 0.
40 TA = 0.

```

C ENTRADA DE LINIA I COMPROVACIONES INICIALES

```

41 DO 20 NL=1,MRE
42 READ(ENT,60) (NL(J,NLIN), J=1,NNUM)
43 DO 12 J=1,NNUM
44 I = NL(J,NLIN)
45 IF (I.GT.NN) GO TO 22
46 .IF (I.NE.0) GO TO 8
47 NF = J - 1
48 GO TO 14
49 8 IF (J.EV.1) GO TO 12
50 MO = J - 1
51 DO 10 M=1,MO
52 IF (I.NE.NL(M,NLIN)) GO TO 10
53 IF (.NOT.(NL(J+1,NLIN).EQ.0.OR.J.EQ.NNUM)) GO TO 24
54 CIRC(NLIN) = .TRUE.
55 NF = J
56 GO TO 14
57 10 CONTINUE
58 12 CONTINUE
59 NF = MNUM
60 N0 = NF - 1
61 NHU(NLIN) = NF

```

C MATRIUS T, NUS I CIRC.

```

62 DO 16 L=1,N0
63 I = NL(L,NLIN)
64 J = NL(L+1,NLIN)
65 IO = MAX(I,J)
66 JO = MIN(I,J)
67 T(L+1,NLIN) = T(L,NLIN) + C(I0,JO)
68 NUS(I,NLIN) = .TRUE.
69 MM = NL(NF,NLIN)
70 NUS(M,NLIN) = .TRUE.
71 TA = TA + T(NF,NLIN)

```

C SEMI-MATRIU D (NOMBRE DE LINIES),

```

72 IF (.NOT.CIRC(NLIN)) GO TO 18
73 NF = NF - 1
74 N0 = N0 - 1
75 DO 20 L=1,N0
76 I = NL(L,NLIN)
77 NO = L + 1
78 DO 20 M=N0,NF
79 K = NL(M,NLIN)

```

\*ID, SEQ.

ISN \* 1 SOURCE STATEMENT / ERROR MESSAGE

```

80 IO = MAX0(I,K)
81 KO = MIN0(I,K)
82 20 D(I0,KO) = D(I0,KO) + 1

```

C ENLLAC AMB EL PROGRAMA PRINCIPAL

```

83 CALL FREQ
84 STOP

```

C MISSATGES D'ERROR

```

85 22 WRITE(10R,86) J, NLIN, I, NN
86 STOP
87 24 WRITE(10R,88) M, J, NLIN, I
88 STOP
89 82 FORMAT('1' // 'ONOMBRE DE NUSOS EXCESSIU,(', I4, ')', EL LIMIT ES',
1 13, ', EL PROGRAMA S'ATURA,')
90 84 FORMAT('1' // 'ONOMBRE DE LINIES EXCESSIU,(', I4, ')', EL LIMIT ES',
1 13, ', EL PROGRAMA S'ATURA,')
91 86 FORMAT('1' // 'DEL NUS', I3, '-E DE LA LINIA', I3, '(', I4, ') ES
1 MES GRAN QUE EL NOMBRE TOTAL DE NUSOS.(', I2, ')', EL PROGRAMA S'IA
1 TURA,')

```

```

92 88 FORMAT('1' // 'ELS NUSOS', I3, '-E I', I3, '-E DE LA LINIA',
1 13, '(', I4, ') SON REPETITS, /
2 'OLA LINIA ES CREUA A SI MATEIXA, /
3 'DEL PROGRAMA S'ATURA, /
END
93

```



### 8.2.5.2. Subrutina FREQ.

Es, de fet, el programa principal, que controla tot l'algorisme, tal com es posa de manifest en el diagrama.

Heus ací una enumeració no gaire detallada de les seves funcions:

- a) lectura de paràmetres d'opció:
  - NREV : periodicitat de les descripcions de la xarxa.
  - ILLIS: indicador de llistat (reduït o complet).
  - IX : " d'assignació de partida (calculada o donada).
- b) assignació de partida dels busos.
  - o bé els llegeix.
  - o bé els assigna proporcionalment al temps de recorregut de la línia, car encara no se sap la demanda absorbida.
- c) descripció de la xarxa inicial.
  - llistat de classificació dels viatges, segons nombre de línies.
  - llistat de paràmetres del procés, amb càlcul previ dels costos unitaris inicial i mínim.
  - llistat de les línies.
- d) crida la subrutina RASS, que s'ocupa de la primera part de l'algorisme (assignació sota hipòtesi de línia única).
- e) càlcul dels vectors de guanys i pèrdues ( $\Delta_i$  i  $\nabla_i$ ) i de la matriu d'interaccions ( $\Pi_{rs}$ ).
- f) recerca del parell de línies (augmentada i disminuïda) que forneixen una disminució de cost màxima.
  - efectuar el canvi d'autobús, amb totes les variacions de dades que comporta.
  - imprimir la informació sobre el canvi.
  - llistar, si correspon, les línies.
  - si hi ha hagut guany, passar a una altra iteració, és a dir, tornar a e). Si no, continuar a g)
- g) final de procés.
  - llistat de les línies.
  - llistat de classificació de viatges segons nombre de línies anàleg a l'inicial.
  - llistat comparat de les assignacions final i inicial de busos.
  - fi de programa.

ISN \* I SOURCE STATEMENT / ERROR MESSAGE \*ID, SEQ

C \*\*\*\*\*

C ASSIGNACIO DE BUSOS I DETERMINACIO DE FREQUENCIES DE LES LINES D'UNA XARXA

C \*\*\*\*\*

```

1 SUBROUTINE FREQ
2 REAL INT, REST(40)
3 INTEGER D, AUT, NBIN(40), DIF(40)
4 INTEGER*4 IW, IP, ID
5 LOGICAL NUS, CIRC
6 COMMON /GROUP 0/ V, Z, TA, AUT, ILLIS
7 /GROUP 1/ NUS(44,40), MRE
8 /GROUP 2/ NL(25,40), T(25,40), NNU(40), CIRC(40), YBUS
9 /GROUP 3/ INT(40,3), NUS(40), E, PE, BETA, HAC
10 /GROUP 4/ INT, NLIT(6), IT(6,8), R(6), TIT(6), YV(7), JOPT
11 /GROUP 5/ D(44,44), C(44,44), NN, NOO
12 /GROUP 6/ W, W1, W2, R1, R2, CM, ID, NORD, NREV
13 /GROUP 7/ G(40), P(40), LAU, LDI
14 /GROUP 8/ H(40,40)
15 EQUIVALENCE (REST(1), NBIN(1)), (REST(21), DIF(1))
16 DATA INT, ISUR /5, 6/
17 DATA HUF6 /32/

```

FACOM

C ASSIGNACIO INICIAL DE BUSOS A LA XARXA

C \*\*\*\*\*

```

10 READ(1,62) NREV, ILLIS, IX
11 IF(IX.LE.0) GO TO 4

```

C ASSIGNACIO DONADA

C \*\*\*\*\*

```

12 READ(1,64) (NBUS(NLIN), NLIN=1,MRE)
13 AUT = 0
14 DO 2 NLIN=1,MRE
15 2 AUT = AUT + NBUS(NLIN)
16 GO TO 10

```

C CALCUL A PARTIR DEL NOMBRE TOTAL DE BUSOS, ASSIGNACIO PROPORCIONAL,

C \*\*\*\*\*

```

17 4 FAUT = FLOAT(AUT)
18 N = 0
19 DO 6 NLIN=1,MRE
20 NF = NNU(NLIN)
21 ZZ = FAUT * T(NF,NLIN) / TA
22 NBUS(NLIN) = ZZ
23 REST(NLIN) = ZZ - FLOAT(NBUS(NLIN))
24 N = N + NBUS(NLIN)
25 NVAR = AUT - N
26 IF(NVAR.EQ.0) GO TO 10
27 DO 8 J=1,NVAR
28 ZZ = 0.
29 DO 7 NLIN=1,MRE
30 IF(REST(NLIN).LE.ZZ) GO TO 7
31 ZZ = REST(NLIN)
32 NOPT = NLIN
33 7 CONTINUE
34 NBUS(NOPT) = NBUS(NOPT) + 1
35 8 REST(NOPT) = 0.

```

\*ID. SEQ.

SOURCE STATEMENT / ERROR MESSAGE

ISN

```

C INTERVALS
C
10 DO 12 NLIN=1,NRE
11   NBN(NLIN) = NBS(NLIN)
12   CALL MINTER(NLIN)
C DESCRIPCIO DE LA XARXA INICIAL
C *****
39   CALL SORTIT
C CALCUL DELS COSTOS EXTREMS I DE LES POBLACIONS
C
40   W1 = 0.
41   W2 = 0.
42   W  = 0.
43   KP = 0.
44   KD = 0.
45   DO 22 I=1,N00
46     NO = I + 1
47     DO 22 K=NO,MN
48       A = D(I,K)
49       IF(A.LE.0.) GO TO 22
50       * * * * A
C COST ACTUAL
C
51   CALL TIRM(I, K, X)
52   W1 = W1 + A * X
53   IF(NIT.NE.0) GO TO 14
54   U = X
55   GO TO 20
56   14 RD = ND + (1.-YY(NIT+1)) * A
C COST MINIM I DEMANDA POTENCIAL
C
57   IF(NIT.GE.2) GO TO 16
58   TIT(I) = R(I) + E
59   CALL MOSI(I, K, U)
60   GO TO 19
61   16 DO 18 JIT=1,NIT
62     TIT(JIT) = R(JIT) + E
63     CALL MOS2(I, K, U)
64     19 RP = RP + (1.-YY(NIT+1)) * A
65     20 W2 = W2 + A * U
66     22 CONTINUE
67     DC 23 I=1,MN
68       W  = W  + FLOAT(D(I,I))
69       IP = W
70       ID = RP
71       W1 = W1 / W
72       W2 = W2 / W
73
C IMPRESSIO DE PARAMETRES I LINES
C
74   SA = TA * YBUS / 1000.
75   WRITE(ISOR,66) NN, MRE, AUT, SX, V, YBUS, PE, E, BETA, HAC,
76   1 W1, IP, ID, W1, W2, NREV
77   WRITE(ISOR,68)

```

ISN \* 1 SOURCE STATEMENT / ERROR MESSAGE \*1D, SEQ.

78 CALL LLIR

C REASSIGNACIO SOTA HIPOTESI DE LINIA UNICA

79 CALL RASS  
80 NRAS = NORD

C CALCUL DELS GUANYES PERDUES I INTERACCIONS.

C \*\*\*\*\*

81 LAU = 0  
82 LDI = 0  
83 DO 27 M=1,MRE  
84 G(M) = 0.  
85 P(M) = 0.  
86 DO 27 N=1,MRE  
87 H(M,N) = 0.  
88 DO 32 I=1,NOD  
89 NC = I + 1  
90 DO 32 K=NO+NN  
91 IF(O(I,K).LE.O) GO TO 32  
92 IF(O(K,I)-1) 32, 28, 30  
93

27

DO 32 I=1,NOD

NC = I + 1

DO 32 K=NO+NN

IF(O(I,K).LE.O) GO TO 32

IF(O(K,I)-1) 32, 28, 30

GO TO 32

30 CALL BLOC2(I,K)

32 CONTINUE

94

95

96

C RECERCA DEL PARELL DE LINIES OPTIM

C \*\*\*\*\*

34 NORD = NORD + 1

GMAX = 0.

DO 35 N=1,MRE

IF(NBUS(N).LE.O) P(N) = -1.E+30

35 CONTINUE

DO 36 M=1,MRE

W = G(M)

DO 36 N=1,MRE

X = W + P(N) + H(M,N)

IF(X.LE.GMAX) GO TO 36

GMAX = X

LAU = M

LDI = N

36 CONTINUE

97

98

99

100

101

102

103

104

105

106

107

108

109

110

C CANVIS A EFECTUAR

C

IF(GMAX.LE.O) GO TO 49

NBUS(LAU) = NBUS(LAU) + 1

NBUS(LDI) = NBUS(LDI) - 1

CALL MINTER(LAU)

CALL MINTER(LDI)

111

112

113

114

115

C DESCRIPCIO DEL DARRER CANVI I DE L'ESTAT DE LA XARXA RESULTANT

IF(MOD(MOD(NORD-NRAS,NREV),NUFE).EQ.1) WRITE(ISOR,74)

CALL COSTMI

WRITE(ISOR,76) NORD, LAU, NBUS(LAU), INT(LAU,2), INT(LAU,1),

LDI, NBUS(LDI), INT(LDI,2), INT(LDI,1),

CA, ID, R1, R2

1

3

C

\*ID, SEQ,

ISM \* 1 SOURCE STATEMENT / ERROR MESSAGE

C DESCRIPCIO DE LA XARXA EN L'ESTAT ACTUAL

```

119 IF(MOD(NORD-NRAS,NREV).EQ.0) GO TO 40.
120 WRITE(LISOR,70) NORD
121 DO 38 NLI=N+1,MRE
122 IF(NBUS(NLIN).NE.0) CALL LLISF(NLIN)
123 38 CONTINUE

```

C ACTUALITZACIO DELS VALORS SUSCEPTIBLES D'HAVER CANVIAT

```

124 40 P(LAU) = -G(LAU)
125 G(LD1) = -P(LD1)
126 G(LAU) = 0.
127 P(LD1) = 0.
128 DO 42 M=1,MRE
129 DO 42 N=1,MRE
130 H(N,N) = 0.
131 DO 48 I=1,N00
132 IN = I + 1
133 DO 48 K=N0,NIN
134 IF(D(K,K).LE.0) GO TO 48
135 JF(D(K,I)-J) 48, 44, 46
136 44 IF(.NOT.(GUS(I,LAU).AND.NBUS(K,LAU)
137 1 .OR. NUS(I,LD1).AND.NUS(K,LD1))) GO TO 48
138 CALL BLOC1(I,K)
139 GO TO 48
140 46 CALL BLOC2(I,K)
141 48 CONTINUE
142 GO TO 34

```

C FINAL DE PROCES

```

142 49 NORD = NORD - 1
143 IF(MOD(NORD-NRAS,NREV).EQ.0) GO TO 32
144 WRITE(LISOR,70) NORD
145 CALL LLIR
146 CALL SORTIT
147 DO 54 NLI=N+1,MRE
148 DIF(NLIN) = NBUS(NLIN) - N5IN(NLIN)
149 WRITE(LISOR,72) (N, NBIN(N), NBUS(N), DIF(N), N=1,MRE)
150 RETURN

```

C FORMATS

```

151 62 FORMAT(13, 77X)
152 64 FORMAT(26I3)
153 66 FORMAT(//, 'PARAMETRES DEL PROCES',
154 1, '*****')
155 2 31X, 'NOMBRE DE ZONES', 17
156 3 31X, 'NOMBRE DE LINES', 17
157 4 31X, 'NOMBRE DE BUSOS', 17
158 5 31X, 'LONGITUD DE XARXA', 17
159 6 31X, 'VELOCITAT A PEU', 17
160 7 31X, 'VELOCITAT DE 50S', 17
161 8 31X, 'PENALITZACIO ESPERA', 17
162 9 31X, 'EQUIVALENT DE LA TARIFA', 17
163 R 31X, 'SENSIBILITAT A LA DIF. DE COSTOS', 17
164 S 31X, 'DIFERENCIA AMB EL COST MINIM', 17
165 A 31X, 'PUBLACIO ACTIVA TOTAL', 17

```

FACOM BOS2 FORTRAN -730216- 0003-03 FREQ

\*ID, SEQ.

ISN \* I SOURCE STATEMENT / ERROR MESSAGE

```

154 T 31X, 'POBLACIO POTENCIAL ABSORBIBLE', 17 /
      B 31X, 'POBLACIO ABSORBIDA PER LA XARXA', 17 /
      C 31X, 'COST INICIAL', F13.5, 1X, 'MIN.', /
      D 31X, 'COST MINIM', F13.5, 1X, 'MIN.', /
      E 31X, 'PERIODICITAT DE LES DESCRIPCIONS', 17, 5X, 'CANVIS', /
155 68 FORMAT('ESTAT INICIAL DE LA XARXA', /
      *, '*****') /
156 70 FORMAT('ESTAT DE LA XARXA DESPRES DEL CANVI', 14 /
      *, '*****') /
157 72 FORMAT('COMPARACIO D'ASSIGNACIONS', /
      *, '*****') /
      1 35X, 2('BUSOS', 5X) /
      2 25X, 'LINIA INICIALS FINALS SALDO' /
      4 40(20X, 4110 /) /
158 74 FORMAT('1, 16X, 'LINIA AUGMENTADA', 16X, 'LINIA DISMINUIDA', /
      1 5X, 2(11X, 'BUSOS INTERVAL', 4X), /
      2 7X, 'COST MITJA DEMANDA', 9X, 'GUANYS', /
      3 5X, 2(18X, 'ACTUAL ANTERIOR'), 20X, 'XARXA PERCENT, ACUMULAT', /
159 76 FORMAT('15, 2(6X, '#, 12, '#, 15, 3X, F6.2, /
      1 F16.5, 19, F11.5, F10.5) /
      1 END

```

### 8.2.5.3. Subrutina RASS.

S'ocupa de la primera part de l'algorisme. Reassigna els autobusos sota la hipòtesi de no interrelació, o bé que entre tot parell de nusos només hi ha una línia. Obté d'una manera molt més ràpida una assignació acceptable. La segona part s'ocupa de millorar-la per bé que molt més lentament.

El procés que segueix és:

- i) càlcul de la demanda absorbida per cada línia, a través de la subrutina DEMLIX.
  - ii) reassignació pròpiament dita, tal com s'explica a la nota teòrica.
  - iii) impressió de la informació relativa a aquesta reassignació.
  - iv) comprovació que la darrera reassignació difereix de l'anterior en, almenys, un bus. Si és així, passar a la següent iteració: anar a ii).
- Si no és així, vol dir que aquesta part ja ha donat de sí tant com podia, i acabar: passar a v).
- v) descripció de la xarxa, i fi de subrutina.

\*ID, SEQ.

SOURCE STATEMENT / ERROR MESSAGE

ISN

C \*\*\*\*\*

C REASSIGNACIO DE BUSOS, SOTA HIPOTESI DE LINIA UNICA

C \*\*\*\*\*

```

1 SUBROUTINE RASS
2   INT, VASS(40), REST(40)
3   INTEGER AUT, NBIN(40), NAUG(40)
4   INTEGER*4 ID, NDEM
5   LOGICAL NUS, CIRC
6   COMMON /GRUP 0/ V, Z, TA, AUT, ILLTS
7   /GRUP 1/ NUS(44,40), NRE
8   /GRUP 2/ NL(25,40), T(25,40), NNU(40), CIRC(40), YBUS
9   /GRUP 3/ INI(40,3), NBUS(40), E, PE, BETA, HAC
6   /GRUP 5/ W, W1, W2, R1, R2, CM, ID, NORD, NREV
9   /GRUP 9/ NDEM(40)
7   EQUIVALENCE (VASS(1), REST(1))
8   DATA ISOR /6/

```

C INICIALIZACIO

C \*\*\*\*\*

```

9 DO 140 NORD=1,50
10 DO 102 NLIN=1,MRE
11 NBI(NLIN) = NBUS(NLIN)
12 IF(NORD.EQ.1) GO TO 122

```

C CALCUL DE LA DEMANDA ABSORBIDA PER LINIA

C \*\*\*\*\*

C CALL DEMLIX

C REASSIGNACIO PROPIAMENT DITTA

C \*\*\*\*\*

```

14 122 SASS = 0.
15 DO 124 NLIN=1,MRE
16 NF = NNU(NLIN)
17 VASS(NLIN) = SORT((NF,NLIN) * FLOAT(NDEM(NLIN)))
18 SASS = SASS + VASS(NLIN)
19 KU = FLOAT(AUT) / SASS
20 N = 0
21 DO 126 NLIN=1,MRE
22 ZZ = KO * VASS(NLIN)
23 NBUS(NLIN) = ZZ
24 REST(NLIN) = ZZ - FLOAT(NBUS(NLIN))
25 N = N + NBUS(NLIN)
26 NVAR = AUT - N
27 IF(NVAR.EQ.0) GO TO 130
28 DO 128 J=1,NVAR
29 ZZ = 0.
30 DO 127 NLIN=1,MRE
31 IF(REST(NLIN).LE.ZZ) GO TO 127
32 ZZ = REST(NLIN)
33 KOPT = NLIN
34 127 CONTINUE
35 NBUS(KOPT) = NBUS(KOPT) + 1
36 128 REST(KOPT) = 0.

```

C CALCULS FINALS I IMPRESSIO DE RESULTATS





#### 8.2.5.4. Subrutina BLOC 1

En la segona part de l'algorisme, es calculen els guanys i pèrdues deguts al canvi de línia d'un autobús. Per això fer, hi ha dues subrutines. La present, BLOC 1, ho fa per a aquells parells de nusos units per una sola línia, és a dir, calcula els vectors de guany i pèrdua ( $\Delta_l$  i  $\nabla_l$ ). La següent, BLOC 2, s'ocuparà de la matriu d'interaccions. ( $\Pi_{rs}$ ).

La raó d'aquesta divisió, rau en un estalvi de temps. Mentre els vectors  $\Delta_l$  i  $\nabla_l$  romanen constants després de cada canvi, la matriu d'interaccions  $\Pi_{rs}$  varia en gran part.

La subrutina BLOC 1, doncs, calcula els vectors al començament de la segona part; a les següents iteracions, comprova si la línia en qüestió ha estat variada en el canvi anterior o no. En el primer supòsit, recalcula.

FACOM UC52 FORTRAN -730216- 0003-03

\*ID, SEQ.

\* 1 SOURCE STATEMENT / ERROR MESSAGE

\*\*\*\*\*

VARIACIO DE COST, AMB UNA SOLA LINIA

\*\*\*\*\*

SUBROUTINE BLOC1(I, K)

REAL INT, U(8)

LOGICAL NUS

COMMON /GROUP 1/ NUS(44,40), MRE

3 /GROUP 3/ INT(40,3), NBUS(40), E, PE, BETA, HAC

4 /GROUP 4/ NIT, NLIT(6), IT(6,8), R(6), TIT(6), YY(7), NLIN

7 /GROUP 7/ G(40), P(40), LAU, LDI

C TEMPS ACTUAL

5 A = LEMA(I,J,K)

6 CALL TIRM(I, K, T)

C GUANY (+1 BUS)

7 IF(NLIN.EQ.LDI) GO TO 102

8 U(1) = INT(NLIN,3)

9 TIT(1) = R(1) + E + PE \* TEME(1,U)

10 CALL MOS1(I, K, TG)

11 G(NLIN) = G(NLIN) + A \* (T-TG)

C PERDUA (-1 BUS)

12 IF(NLIN.EQ.LAU) RETURN

13 102 U(1) = INT(NLIN,1)

14 TIT(1) = R(1) + E + PE \* TEME(1,U)

15 CALL MOS1(I, K, TP)

16 P(NLIN) = P(NLIN) + A \* (T-TP)

17 RETURN

18 END

#### 8.2.5.5. Subrutina BLOC 2.

Calcula la matriu d'interaccions  $\Pi_{rs}$  després de cada canvi.

Al seu torn, té dues parts, tal com es descriu a la nota teòrica.

- i) calcula els guanys/pèrdues soferts per un parell de nusos, quant la línia augmentada com la disminuïda els uneixen.
- ii) calcula els guanys/pèrdues, quan només una de les línies uneix el parell.

Tal com es desprèn del capítol 3, en i), s'afecta un element de i a ii), s'afecta una fila i una columna.



SOURCE STATEMENT / ERROR MESSAGE

ISN \* I

\*ID, SEQ.

222 IF(LIN,NE,LDI) GO TO 224

U(L) = INT(LIN,1)

GO TO 226

224 U(L) = INT(LIN,2)

226 CONTINUE

228 TIT(J) = R(J) + E + PE \* TEME(NPRO,U)

CALL MOS2(I, K, S)

C AFECTACIO DEL GUANY O DE LA PERDUA A LA MATRIU D'INTERACCIONS

H(LAU,LDI) = H(LAU,LDI) + A \* (T - S)

232 CONTINUE

C CALCUL DE LES VARIACIONS SENSE INTERACCIO

C \*\*\*\*\*

DO 246 M=1,MRE

IF(.NOT.,PERT(M)) GO TO 246

DO 238 J=1,NIT

NPRO = NLIT(J)

DO 236 L=1,NPRO

LIN = IT(J,L)

IF(LIN,NE,M) GO TO 234

U(L) = INT(LIN,3)

V(L) = INT(LIN,1)

GO TO 236

234 U(L) = INT(LIN,2)

V(L) = U(L)

236 CONTINUE

TIT(J) = R(J) + E + PE \* TEME(NPRO,U)

TIV(J) = R(J) + E + PE \* TEME(NPRO,V)

CALL MOS2(I, K, SAU)

ASAU = A \* (T - SAU)

DO 240 J=1,NIT

TIT(J) = TIV(J)

CALL MOS2(I, K, SDI)

ASDI = A \* (T - SDI)

C AFECTACIO DEL GUANY O DE LA PERDUA A LA MATRIU D'INTERACCIONS

DO 244 N=1,MRE

IF(PERT(N)) GO TO 244

H(N,M) = H(M,N) + ASAU

H(N,M) = H(N,M) + ASDI

244 CONTINUE

246 CONTINUE

RETURN

END

#### 8.2.5.6. Subrutina SORTIT

Calcula i imprimeix el llistat de classificació dels fluxos segons el nombre de línies.

Recorre la matriu D, meitat inferior,  $[\eta_{ij}]$ , on hi ha allotjats el nombre de línies que uneix cada parell. Segons aquest valor i el temps de recorregut en cada itinerari, calcula la proporció de demanda absor**u**bida per tots ells i la no absorbida.

Aquests valors, són acumulats en una taula i impresos.





\*ID, SEQ,

SOURCE STATEMENT / ERROR MESSAGE

ISN

```

38 DO 316 L=1,9
39 ITOT = ITOT + IDEM(L)
40 DO 316 N=1,3
41 316 TOT(N) = TOT(N) + DEM(L,N)

```

C CONVERGIO A ENTERS, PER TAL D'IMPRIMIR-LOS

```

42 DO 318 N=1,3
43 NTOI(N) = TOT(N)
44 DO 318 L=1,9
45 318 NDEM(L,N) = DEM(L,N)

```

C IMPRESSIO DE RESULTATS

```

46 WRITE(ISOR,352) (NUM(L), IDEM(L), (NDEM(L,N), N=1,3), L=1,9),
1 ITOT, (NTOT(N), N=1,3)
47 RETURN

```

C FORMATS

```

48 352 FORMAT('RELACIO DE LES UNIONS ENTRE PARELLS DE NUSOS',
1 '*****')
2 88X, 'DEMANDES',
3 31X, 'NO.LINIES', 10X, 'NO.PARELLS DE NUSOS', 9X, 'NO ABSORBIDA',
4 ABSORBIDA
5 9(140, 2125, 2112 /)
6 24X, 'TOTALS', 30X, 15, 125, 2112, 3
49 END

```

#### 8.2.5.7. Subrutina COSTMI.

Calcula el temps mitjà i la demanda total absorbida per la xarxa. Recorre la matriu de demandes, i per a cada flux en calcula els diversos temps de viatge.

Finalment, troba quin és el guany percentual obtingut en el darrer canvi, així com l'acumulat fins al moment.

FACOM BOS2 FORTRAN -730216- 0003-03

\*ID, SEQ.

\*\*\*\*\* SOURCE STATEMENT / ERROR MESSAGE \*\*\*\*\*

C COST MITJA I DEMANDA ABSORBIDA PER LA XARXA

SUBROUTINE COSTMI

INTEGER D

INTEGER\*4 ID

COMMON /GRUP 4/ NIT, NLI(6), IT(6,8), R(6), III(6), YY(7), JOPT

/GRUP 5/ D(4,44), C(4,44), NN, NOO

/GRUP 6/ W, W1, W2, R1, R2, CM, ID, NORD, NREV

CM = 0.

S = 0.

DO 408 I=1,NOO

NO = I + 1

DO 408 K=NO,NN

DM = D(I,K)

IF(DM.LE.0.) GO TO 408

CALL TIRM(I, K, X)

CM = CM + X \* DM

S = S + (1. - YY(NIT+1)) \* DM

408 CONTINUE

CM = CM / W

ID = S

C GUANYNS PERCENTUAL I ACUMULAT

IF(NORD.EQ.1) CANT = W1

R1 = (CANT-CM) / W2 \* 100.

R2 = (W1 - CM) / W2 \* 100.

CANT = CM

RETURN

END

#### 8.2.5.8. Subrutina DEMLIX.

Calcula la demanda o càrrega de cada línia.

Poden presentar-se diversos graus de complexitat:

- 0) No hi ha cap línia: la demanda no s'assigna enlloc.
- 1) 1 sola línia: el flux que no va a peu, s'hi assigna.
- 2) 2 ò més línies, amb una sola línia per itinerari: el flux assignat a cada itinerari, es carrega a la línia que el constitueix.
- 3) 2 ò més línies, amb itineraris servits per més d'una línia: a més de l'assignació per itineraris, tal com s'ha fet al cas anterior, cal repartir la demanda entre les diverses línies de cada itinerari, proporcionalment a llur freqüència.



FACOM BOS2 FORTRAN -730216-0003-03 DEMLIX

\*1 SOURCE STATEMENT / ERROR MESSAGE #10, SEQ.

ISM \* 1 SOURCE STATEMENT / ERROR MESSAGE

33 414 DEM(JOPT) = DEM(JOPT) + YY(1) \* A  
34 416 CONTINUE

C CONVERSIO A ENTER

35 DO 418 NLIN=1,MRE  
36 418 NDEM(NLIN) = DEM(NLIN)  
37 RETURN  
38 END

#### 8.2.5.9. Subrutina LLIR.

S'ocupa, en general, dels llistats particulars de cada línia. Cas que es demani un llistat complet, crida la subrutina LLISF , per a totes aquelles línies que tinguin almenys un bus.

Cas de demanar un llistat reduït, ella mateixa el forneix. El seu contingut s'ha explicat en parlar de les sortides del programa.

\*ID, SEQ.

SOURCE STATEMENT / ERROR MESSAGE

ISN

```

C *****
C LLISTAT REDUIT DE LA XARXA
C *****
C
1 SUBROUTINE LLIR
2 INTEGER D
3 INTEGER*4 NDEM
4 REAL INT, S(40)
5 LOGICAL NUS, CIRC
6 COMMON /GRUP 0/ V, Z, TA, AUT, ILLIS
1 /GRUP 1/ NUS(44,40), MRE
2 /GRUP 2/ NLC(25,40), T(25,40), NNU(40), CIRC(40), YBUS
3 /GRUP 3/ INT(40,3), NBUS(40), E, PE, BETA, HAC
5 /GRUP 5/ D(44,44), C(44,44), NN, NOO
9 /GRUP 9/ NDEM(40)
7 DATA ISOR /6/
C
C CAS DE LLISTAT COMPLET
C
8 IF(ILLIS.EQ.0) GO TO 504
9 DO 502 NLIN=1,MRE
10 IF(NBUS(NLIN),NE.0) CALL LLISF(NLIN)
11 502 CONTINUE
12 RETURN
C
C CALCUL DE LA DEMANDA ABSORBIDA PER CADA LINIA
C
13 504 CALL DEMLIX
C
C LONGITUD DE LA XARXA
C
14 DO 506 NLIN =1,MRE
15 NF = NNU(NLIN)
16 506 S(NLIN) = YBUS * T(NF,NLIN) / 1000.
C
C IMPRESSIO DE VALORS
C
17 WRITE(ISOR, 552) (J, NNU(J), S(J), NDEM(J), INT(J,2), NBUS(J), J,
18 J=1,MRE)
19 RETURN
552 FORMAT('01, 30X, ' LINIA NUSOS LONGITUD DEMANDA INTERV
1AL BUSOS LINIA' /
2 56X, '(KM,)', 14X, '(MIN,)' /
2 40(31X, 2110, F10.2, 110, 4X, F6.2, 2110. /)
END
20

```



## 8.2.5.10 Subrutina LLISF.

Calcula i imprimeix el llistat complet relatiu a una línia.

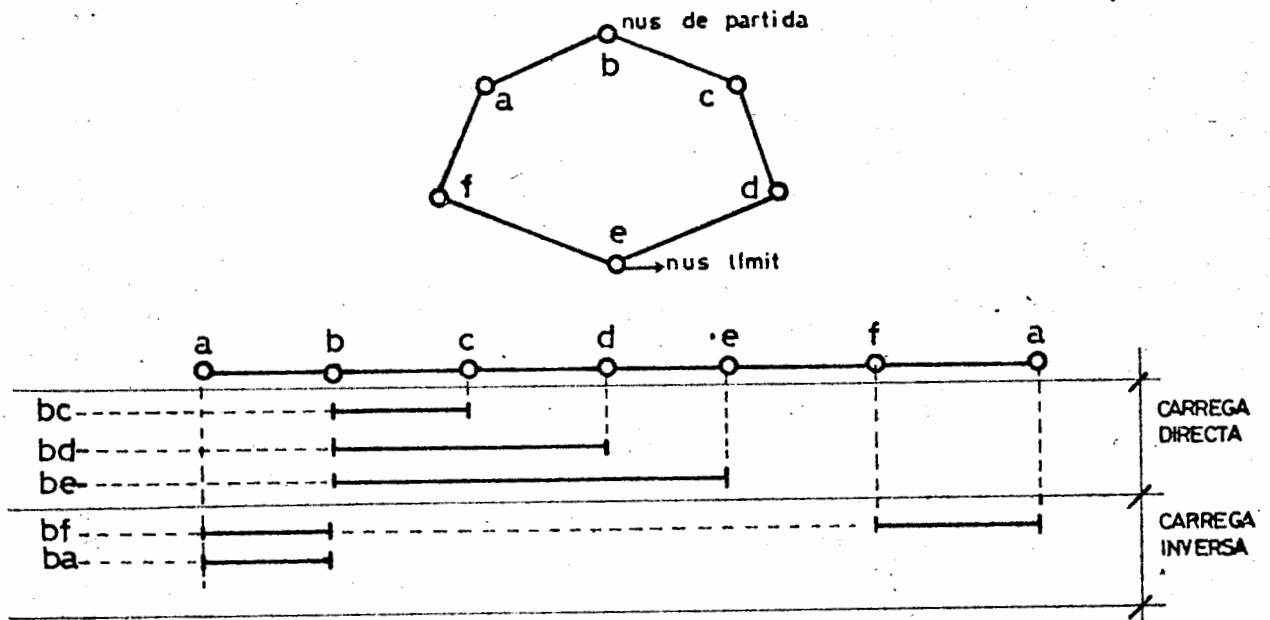
La subrutina es desglossa en tres parts:

## a) inicialització:

- comptadors de càrrega de la línia posats a 0.

- recerca del punt de canvi, (només per a línies circulars).

quan es tracta d'assignar la càrrega entre dos nusos que pertanyen a una línia circular, se segueix el criteri de fer-ho pel camí més curt. Donat que, sota el punt de vista informàtic, una línia circular és un enfilall de nusos on el primer i el darrer coincideixen, cal trobar quin és el nus límit a partir del qual la



Trams carregats

FIG. 8.2.4

càrrega deixa de ser directa per passar a ser inversa. El significat donat als termes directa i inversa, queda explicat en la figura 8.2.4.

## b) càlcul de la demanda total i per tram.

Per un mètode anàleg a l'emprat a la subrutina DEMLIX, es calcula quina és la càrrega que la línia en qüestió suporta. Tot seguit, l'assigna als trams intermedis entre els nusos. Si la línia és circular i cal fer càrrega inversa, assigna els trams extrems.

## c) determinació de la situació del llistat dintre de la pàgina.

Per tal d'obtenir un llistat compacte ensems que llegible, cal enquibir un màxim de línies a cada pàgina.

Per això fer, aquesta part de la subrutina, abans d'imprimir cada línia, comprova si té espai suficient a la pàgina en curs o li cal saltar a la següent.

ISN \* 1 SOURCE STATEMENT / ERROR MESSAGE \*ID, SEQ.

\*\*\*\*\*  
IMPRESSIO DE LA LINIA  
\*\*\*\*\*

```

1 SUBROUTINE LLISF(NLIN)
2 REAL INT
3 INTEGER LIM(25)
4 INTEGER*4 CAR(25), KAR, NDEM
5 LOGICAL NUS
6 COMMON /GRUP 2/ NL(25,40), T(25,40), NNU(40), CIRC(40), YBUS
   /GRUP 3/ INT(40,3), NBUS(40), E, PE, BETA, HAC
   /GRUP 4/ NIT, NLIT(6), IT(6,8), R(6), TIT(6), YY(7), JOPT
   /GRUP 9/ NDEM(40)
7 DATA NTF, NTRAM / 69, 15 /, ISOR / 6/

```

FACOM

INICIALITZACIO

```

8 NF = NNU(NLIN)
9 N0 = NF - 1
10 NDE(M(NLIN)) = 0
11 DO 502 J=1,N0
12 502 CAR(J) = 0

```

PUNTS DE CANVI DE CARRERA PER A LINES CIRCULARS

```

13 IF(.NOT.CIRC(NLIN)) GO TO 509
14 SEMI = 0.5 * T(NF,NLIN)
15 NF = NF - 1
16 N0 = N0 - 1
17 DO 508 L=1,N0
18 X = T(L,NLIN)
19 N0 = L + 1
20 GO 504 M=NO,NF

```

IF SEMI.LT.(M(NLIN) - X) GO TO 506

504 CONTINUE

LIM(L) = NF + 1

GO TO 506

508 LIM(L) = NF

508 CONTINUE

CALCUL DE LA DEMANDA TOTAL I PER TRAM

```

27 DO 505 L=1,N0
28 I = NL(L,NLIN)
29 N0 = L + 1
30 DO 528 M=NO,NF
31 K = NL(M,NLIN)
32 CALL TRMCI, K, X)

```

CAS DE 2 O MES LINES PER ITINERARI

```

33 IF(NIT-EG.1 .AND. NLIT(1)-EG.1) GO TO 515
34 CG 510 JIT=1,NIT
35 NPRO = NLIT(JIT)
36 DO 510 MIT=1,NPRO
37 IF(CJIT-MIT)-EG.NLIN) GO TO 512
38 510 CONTINUE

```

FACOM BOSZ FORTRAN -730216- 0003-03 LLISF

ISN \* I SOURCE STATEMENT / ERROR MESSAGE \*ID, SEQ.

39 512 IF(NPRO.EQ.1) GO TO 516

C PROPORCIO DE DEMANDA ABSORBIDA PER LA LINIA EN FUNCIO DE LA FRECUENCIA

40 S = 0.

DO 514 MIT=1,NPRO

MM = IT(JITAMIT)

514 S = S + 1. / INT(MM\*2)

S = 1. / (S \* INT(NLIN,2))

KAR = S \* FLOAT(LEMA(1,I,K))

GO TO 518

C CAS DE LINIA UNICA

513 JIT = 1

516 S = 1.

518 KAR = YY(JIT) \* S \* FLOAT(LEMA(1,I,K))

C CARREGA DE LA LINIA

C \*\*\*\*\*

50 \*NDEM(NLIN) = NDEM(NLIN) + KAR  
51 IF(CIRC(NLIN).AND.M.GE.LIM(L)) GO TO 522

C CAS DE LINIA DRETA O CIRCULAR AMB CARREGA DIRECTA

MO = M - 1

DO 520 J=1,MO

520 CAR(J) = CAR(J) + KAR

GO TO 528

C CAS DE LINIA CIRCULAR AMB CARREGA INVERSA

522 DO 524 J=1,NF

524 CAR(J) = CAR(J) + KAR

IF(L.EQ.1) GO TO 528

MO = L - 1

DO 526 J=1,MO

526 CAR(J) = CAR(J) + KAR

528 CONTINUE

IF(.NOT.CIRC(NLIN)) GO TO 529

NF = NF + 1

NO = NO + 1

C DETERMINACIO DE LA SITUACIO DEL LLISTAT

C \*\*\*\*\*

529 IF(NLIN.EQ.1) GO TO 531

MO = NLIN - 1

DO 530 M=1,MO

IF(NDEM(M).NE.0) GO TO 532

CONTINUE

531 LL = NTF - 3

532 IF(NF-NTRAM) 533, 533, 534

533 LO = 7

GO TO 536

534 LC = 10

538 IF(LL-LO) 538,540, 540

538 LL = MTF + 1 - LO

WRITE(150R,554) NLIN

GO TO 542

66

67

68

69

70

71

72

73

74

75

76

77

78

79

\*ID, SEQ.

SOURCE STATEMENT / ERROR MESSAGE

FACOM BOS2 FORTRAN -730216- 0003-03

\*ID, SEQ.

SOURCE STATEMENT / ERROR MESSAGE

540 LL = LL - LO

WRITE(ISOR,556) NLIN

C IMPRESSIO DE LES FRANGES

542 KF = MINO(NQ,NTRAM)

KO = KF + 1

S = YBUS \* T(NF,NLIN) / 1000.

WRITE(ISOR,558) NDEM(NLIN), (J, J=1,KO)

WRITE(ISOR,560) NBUS(NLIN), (CAR(J), J=1,KF)

WRITE(ISOR,562) INT(NLIN,2), (NL(J,NLIN), J=1,KO)

WRITE(ISOR,564) S

IF(NQ.LE.NTRAM) RETURN

KC = NTRAM + 1

WRITE(ISOR,566) (J, J=KC,NF)

WRITE(ISOR,568) (CAR(J), J=KC,NQ)

WRITE(ISOR,570) (NL(J,NLIN), J=KC,NF)

RETURN

C FORMATS

C \*\*\*\*\*

554 FORMAT('ILINIA \*', I2, '(\*)')

556 FORMAT('OLINIA \*', I2, '(\*)')

558 FORMAT('O DEM, ', I5, ' ORD,', I2, ' 1817)

560 FORMAT(' BUSOS, ', I5, ' CAR,', I2, ' 1817)

562 FORMAT(' INT, ', I5, '2, ' NUS, ', I2, ' 1817)

564 FORMAT(' LONG, ', I5, '2)

566 FORMAT('13X, ' ORD,', I2, ' 1817)

568 FORMAT('13X, ' CAR,', I2, ' 1817)

570 FORMAT('13X, ' NUS,', I2, ' 1817)

END

13 ILLEGAL TYPE ( ) NOT ( R )

51 ILLEGAL TYPE ( R ) AND ( L )

63 ILLEGAL TYPE ( ) NOT ( R )

FT610Y

FT610Y

FT610Y

#### 8.2.5.11. Subrutina MINTER.

Per a una línia dada, calcula i emmagatzema els intervals que li pertocuen amb

- el nombre actual de busos.
- un bus menys dels que té assignats.
- un bus més dels que té assignats.

L'aplicació de la subrutina és evident: estalvia de calcular els intervals a cada càlcul que hom fa dintre d'un canvi donat.

FACOM BOS2 FORTRAN -730216- 0003-03

\*JD, SEQ.

ISN \* I SOURCE STATEMENT / ERROR MESSAGE

\*\*\*\*\*

CALCUL DELS INTERVALS D'UNA LINIA

\*\*\*\*\*

1 SUBROUTINE MINTER(NLIN)

2 REAL INT

3 LOGICAL CIRC

4 COMMON /GROUP 2/ NL(25,40), T(25,40), NNU(40), CIRC(40), YBUS

5 3 /GROUP 3/ INT(40,3), NBUS(40), E, PE, BETA, HAC

6 N = NBUS(NLIN) - 2

7 NF = NNU(NLIN)

8 Z = T(NF,NLIN)

9 DO 604 J=1,3

10 M = N + J

11 IF(M.GT.0) GO TO 602

12 INT(NLIN,J) = 1,E+20

13 GO TO 604

14 602 INT(NLIN,J) = 2. \* Z / FLOAT(M)

15 604 CONTINUE

16 RETURN

END

#### 8.2.5.12. Subrutina ITIN.

Per a un flux unit per més d'una línia, determina

- . el nombre d'itineraris.
- . el nombre de línies de cada itinerari.
- . el temps de recorregut de cada itinerari.

La identificació d'itineraris té lloc a través de comparar els temps de recorregut per les línies. Si coincideixen, s'admet que les dues línies comprenen els mateixos nusos entre el parell considerat; és a dir, pertanyen a un mateix itinerari.

ISN \* 1 SOURCE STATEMENT / ERROR MESSAGE \*1D, SEQ.

\*\*\*\*\*

DETERMINACIO D'ITINERARIS I LINES ENTRE 2 ZONES

\*\*\*\*\*

```

1 SUBROUTINE ITIN(I,K)
2 LOGICAL NUS
3 COMMON /GRUP 1/ NUS(44,40), MRE
4 /GRUP 4/ NIT, NLIT(6), IT(6,8), R(6), IIT(6), YY(7), JOBT
5 DATA E / 1.E-3/

```

C RECERCA DE LINES QUE UNEIXEN I AMB K

```

5 L = LEMAC(I,K)
6 N = 0
7 DO 712 NLIN=1,MRE
8 IF(.NOT.(NUS(I,NLIN).AND,NUS(K,NLIN))) GO TO 712
9 N = N + 1
10 IF(N.GT.1) GO TO 702
11 RLIN = DOBUS(NLIN,I,K)
12 GO TO 706
13 NIT = I

```

C PRIMERA LINEA D'UN ITINERARI

```

14 C.C.
15 DO 704 J=1,NIT
16 IF(ABS(RLIN-R(J)),LE.E) GO TO 708
17 704 CONTINUE
18 NIT = NIT + 1
19 708 NLIT(NIT) = 1
20 R(NIT) = RLIN
21 IT(NIT,1) = NLIN
22 GO TO 710

```

C LINEA AMB ITINERARI JA DETECTAT

```

22 C.C.
23 708 NLIT(J) = NLIT(J) + 1
24 MM = NLIT(J)
25 IT(J,MM) = NLIN

```

C TEST DE FINAL

```

25 C.C.
26 710 IF(N.GE.L) RETURN
27 712 CONTINUE
28 RETURN
29 END

```



### 8.2.5.13. Subrutina TIRM.

Es una subrutina fonamental dintre del programa.

El diagrama de jerarquies manifesta que, dels 14 mòduls restants:

- . 6 subrutines la criden directament.
- . 4 " " són cridades per ella;

és doncs, la subrutina central del sistema.

Forneix:

- . el temps de viatge per cada itinerari, considerant la marxa a peu com un itinerari més.
- . la repartició de càrregues entre itineraris.
- . l'itinerari de cost mínim.
- . el cost mitjà de viatge entre el parell de nusos considerat

Consta de 4 parts, segons el nombre de línies:

a) cap línia.

Dóna el cost a peu, que coincideix amb el cost mitjà.

b) una sola línia.

c) diverses línies i diversos itineraris.

En ambdós casos es calcula el temps de viatge i la repartició de càrregues tal com s'ha explicat al punt 5.2.4. en parlar de la funció M.

d) diverses línies en un sol itinerari.

La repartició, en aquest cas, es fa com si hi hagués una sola línia.

ISN \* 1 SOURCE STATEMENT / ERROR MESSAGE \*ID, SEQ.

\*\*\*\*\*

DETERMINACIO D'ITINERARIS, TEMPS ASSOCIAT I REPARTICIO DE CARRERA

\*\*\*\*\*

```

1 SUBROUTINE TIRMI(I, K, CU)
2 INTEGER D
3 REAL INT, UC(8)
4 LOGICAL
5 CORRICH /GRUP 1/ NUS(44,40), MRE
3 /GRUP 3/ INT(40,3), NBUS(40), E, PE, BETA, HAC
4 /GRUP 4/ NIT, NLIT(6), IT(6,8), R(6), TIT(6), YY(7), JOPT
5 /GRUP 5/ D(44,44), C(44,44), NN, NOO

```

C MARXA A PEU

```

6 IO = MINO(I,K)
7 KO = MAXO(I,K)
8 P = C(IU,KO)

```

C SEPARACIO SEGONS EL NOMBRE DE LINES

IF (LEMA(2,I,K)-1) 802, 804, 810

C CAP LÍNIA

\*\*\*\*\*

```

10 802 YY(1) = 1.
11 NIT = 0
12 CU = P
13 RETURN

```

C UNA SOLA LÍNIA

\*\*\*\*\*

```

14 804 DO 805 I=1,MRE
15 IF (NUS(I,NLIN),AND,NUS(K,NLIN)) GO TO 808
16 805 CONTINUE

```

C TEMPS

```

17 808 UC(1) = INT(NLIN,2)
18 JOPT = NLIN
19 NIT = 1
20 R(1) = DOBUS(JOPT, I, K)
21 TIT(1) = R(1) + E + PE * TEME(1,U)
22 ENTRY MOSI(I, K, CU)
23 T = TIT(1)

```

C REPARTICIO DE CARREGUES

```

24 809 X = AMINI(P,T) - HAC
25 S = 1. / (1./T-X) + 1./P-X)
26 YY(1) = S / (T-X)
27 YY(2) = S / (P-X)
28 CU = YY(1) * T + YY(2) * P
29 RETURN

```

C DIVERSES LINES

\*ID. SEQ.

SOURCE STATEMENT / ERROR MESSAGE

ISN # I

C \*\*\*\*\*

30 810 CALL ITIN(I, K)

31 DO 814 J=1,NIT

32 MU = NLIT(J)

33 DO 812 M=1,MO

34 MM = IT(J,M)

35 812 U(M) = INT(MM,2)

36 814 TIT(J) = R(J) + E + PE \* TENE(MO,U)

37 ENTRY MOS2(I, K, CU)

38 IF(NIT.EQ.1) GO TO 822

39 X = P

40 JOPT = NIT + 1

41 DO 816 J=1,NIT

42 IF(TIT(J).GE.X) GO TO 816

43 X = TIT(J)

44 JOPT = J

45 816 CONTINUE

46 X = X - HAC

C REPARTICIO DE CARGUES

47 S = 1. / (P-X)

48 DO 818 J=1,NIT

49 818 S = S + 1. / (TIT(J) - X)

50 S = 1. / S

C TEMPS

51 YY(NIT+1) = S / (P-X)

52 CU = YY(NIT+1) \* P

53 DO 820 J=1,NIT

54 YY(J) = S / (TIT(J)-X)

55 820 CU = CU + YY(J) \* TIT(J)

56 RETURN

C DIVERSES LINES EN UN SOL ITINERARI (ASSIMILAT A UNA SOLA LINEA)

C \*\*\*\*\*

57 822 I = TIT(I)

58 JOPT = 2

59 IF(T.LT.P) JOPT = 1

60 GO TO 809

61 END

#### 8.2.5.14. Funció DOBUS.

Per a una línia i un parell donat, forneix el temps de recorregut, per mitjà d'una consulta a la matriu  $[t_{ik}]$  .  
En les línies circulars , dóna el camí més curt dels dos possibles.



8.2.5.15. Funció LEMA.

Dóna adés la demanda total entre un parell donat, (matriu  $[d_{ik}]$  ), adés el nombre de línies que el serveixen (matriu  $[\eta_{ij}]$  ), per mitjà d'una consulta a la matriu.

ISN \* 1 SOURCE STATEMENT / ERROR MESSAGE \*10, SEQ.

\*\*\*\*\*  
ACCES A LA MATRIU DE DEMANDA  
\*\*\*\*\*

1 FUNCTION LEMA(N, IO, KO)  
2 INTEGER D  
3 COMMON /GRUP 5/ D(44,44), C(44,44), NN, NOO

C DETERMINACIO DELS SUBINDEXS

4 GO TO (1102, 1104), N  
5 1102 I = MINO(IO,KO)  
6 K = MAXO(IO,KO)  
7 GO TO 1110  
8 1104 I = MAXO(IO,KO)  
9 K = MINO(IO,KO)

C CALCUL DEL VALOR

10 1110 LEMA = D(I,K)  
11 RETURN  
12 END

#### 8.2.5.16. Funció TEME.

Pensada per tal de servir deslligada de la resta del programa, és l'única on no apareix la instrucció COMMON.

Donat un cert nombre de línies, fins a 8, i llurs intervals, forneix el temps mitjà d'espera, per aplicació de la fórmula de LAMPKIN i SAALMANS, descrita al punt 5.1.

El mètode de càlcul que s'ha adoptat, es desenvolupa en parlar de l'aa dequació operativa; en resum, ordena creixentment els intervals, i calcula en primer lloc el temps d'espera si només hi hagués les dues primeres línies ( $W_2$ ). Després, afegeix a aquest valor els increments ( $h_3, h_4, \dots, h_7$ ) que tindrien lloc per successiva addició de noves línies.



```

* I SOURCE STATEMENT / ERROR MESSAGE *ID, SEQ.
*****
TEMPS MITJA D'ESPERA
*****
FUNCTION TEME(N, U)
REAL U(8)
C CAS D'UNA SOLA LINIA. CALCUL DIRECTE
IF(N.EQ.1) GO TO 1206.
C ORDENACIO CREIXENT D'INTERVALS
NA = N - 1
DO 1202 ITER=1,NA
  IF1 = N - ITER
  DO 1202 I=1,IF1
    IF(U(I).LE.U(IF1+1)) GO TO 1202
    A = U(I)
    U(I) = U(IF1+1)
    U(IF1+1) = A
  1202 CONTINUE
C CAS DE 8 LINIES, REDUCCIO A 7
IF(N.GT.7) N = 7
C CAS D'INTERVAL INFINIT. LINIA, IGNORADA
1204 IF(N.LE.1.OR.U(N).LT.1.E+19) GO TO 1205
  N = N - 1
  GO TO 1204
C CALCUL EN CADA CAS
*****
1205 GO TO (1206, 1208, 1210, 1210, 1210, 1210, 1210),N
C 1
1206 TEME = U(1) * 0.5
  RETURN
C 2
1208 TEME = U(1) * (0.5 - U(1) / (6.*U(2)))
  RETURN
C 3
1210 U1 = U(1)
  U2 = U(2)
  U3 = U(3)
  TEME = U1 * (0.5 - U1 * ((1./U2 + 1./U3) / 6.
  1 IF(N.LE.3) RETURN
C 4

```

ISN 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26

ISN \* 1 SOURCE STATEMENT / ERROR MESSAGE \*ID. SEQ.

27 U4 = U(4)  
 28 TEME = TEME - U1 \* U1 / (2. \* U4) \* (0.333333333  
 1 - U1 \* ((1./U2 + 1./U3) / 6.  
 2 - U1 / (U2 \* U3 \* 10.))  
 29 IF(N.LE.4) RETURN

30 U5 = U(5)  
 31 TEME = TEME - U1 \* U1 / (2. \* U5) \* (0.333333333  
 1 - U1 \* ((1./U2 + 1./U3 + 1./U4) / 6.  
 2 - U1 \* ((1./U2\*U3) + 1./U4) + 1./U4) / 10.  
 3 - U1 / (U2 \* U3 \* U4 \* 15.))  
 32 IF(N.LE.5) RETURN

33 U6 = U(6)  
 34 TEME = TEME - U1 \* U1 / (2. \* U6) \* (0.333333333  
 1 - U1 \* ((1./U2 + 1./U3 + 1./U4 + 1./U5) / 6.  
 2 - U1 \* ((1./U2\*U3) + 1./U4) + 1./U4) + 1./U4) / 10.  
 3 + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 10.  
 4 - U1 \* ((1./U2\*U3\*U4) + 1./U4) + 1./U4) + 1./U4) / 15.  
 5 + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 15.  
 6 - U1 / (U2 \* U3 \* U4 \* U5 \* 21.))  
 35 IF(N.LE.6) RETURN

36 U7 = U(7)  
 37 TEME = TEME - U1 \* U1 / (2. \* U7) \* (0.333333333  
 1 - U1 \* ((1./U2 + 1./U3 + 1./U4 + 1./U5 + 1./U6) / 6.  
 2 - U1 \* ((1./U2\*U3) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 10.  
 3 + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 10.  
 4 + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 10.  
 5 + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 10.  
 6 - U1 \* ((1./U2\*U3\*U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 15.  
 7 + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 15.  
 8 + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 15.  
 9 + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 15.  
 A - U1 \* ((1./U2\*U3\*U4\*U5) + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 21.  
 B + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 21.  
 C + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) + 1./U4) / 21.  
 D - U1 / (U2 \* U3 \* U4 \* U5 \* U6 \* 28.))  
 38 RETURN  
 39 END

CCUPB/BIBLIOTECA  
 N.º REGISTRO: 193  
 FEGHA: 77.06.01