

17/77

201

UNIVERSITÉ D'ALGER

ECOLE NATIONALE POLYTECHNIQUE

ELECTRONIQUE

DÉPARTEMENT ELECTRICITÉ

THESE DE FIN D'ÉTUDES

المدرسة الوطنية للعلوم التطبيقية  
الجزائر  
SIMULATION D'UN RÉSEAU DE TRANSPORT  
EN COMMUN, APPLICATION AU CALCUL D'UN  
RESEAU OPTIMAL



Proposée par

M.D. OYHANTO

Etudiée par

M.M. E.A. DEKKICHE

M. BOUKERT

Promotion : JANVIER 1977

STIRLING

and John

ESTER

Stirling

UNIVERSITÉ D'ALGER

---

ECOLE NATIONALE POLYTECHNIQUE

---

DÉPARTEMENT ELECTRICITÉ

THESE DE FIN D'ÉTUDES

**SIMULATION D'UN RÉSEAU DE TRANSPORT  
EN COMMUN, APPLICATION AU CALCUL D'UN  
RESEAU OPTIMAL**

Proposée par

**M.D. OYHANTO**

Etudiée par

**M.M. E.A. DEKKICHE**

**M. BOUKERT**

Promotion : JANVIER 1977

à mes parents

à mes amis

Boukert Mustapha

à mes parents

à mes frères et soeurs

à mes amis

Dekkiche El Abbas

" La science remplace du visible compliqué par de l'invisible simple "

Jean Perrin

" Si, depuis qu'elle existe, l'humanité avait construit sa connaissance du monde selon un nominalisme rigoureux, en se contentant de mettre un ordre logique dans les données empiriques, et si la Logique était elle même donnée empiriquement comme une forme naturelle d'organisation de la connaissance, alors nos théories n'auraient jamais exprimé que l'expérience passée et voir l'expérience d'aujourd'hui n'y introduirait aucun arbitre spéculatif ".

André Régnier

" Les infortunes de la raison ".

- A V A N T P R O P O S -

Les transports urbains constituent l'un des problèmes multiples que rencontre l'homme de la grande ville.

La résolution de ce type de problèmes ne peut se faire que par des méthodes scientifiques d'autant plus que leurs caractéristiques sont généralement aléatoires.

Nous présentons dans la présente étude un modèle mathématique qui peut s'avérer utile pour des études éventuelles sur la régulation, sur une modification des itinéraires d'un réseau,...

Cette étude n'aurait pas été possible sans l'aide et les conseils précieux de M. Daniel Oyhanto auquel nous devons une profonde reconnaissance.

Nous remercions le personnel du Centre de Calcul du C.E.R.I (Centre d'études et de Recherche en Informatique) de nous avoir facilité la tâche lors des opérations de programmation.

Nos remerciements vont également à tous les professeurs du Département Electricité, ainsi qu'à tous nos camarades et amis qui nous ont aidé dans la réalisation de ce projet.

# S O M M A I R E

I -	I N T R O D U C T I O N	page	I
II -	L A S I M U L A T I O N		5
	II - 1. Présentation du problème		5
	II - 2. Description du processus		6
	II - 3. Choix des variables et paramètres		8
	II - 4. Le modèle de simulation		14
III -	L A P R O G R A M M A T I O N		27
	III - 1. Programme de simulation		27
	III - 2. Principaux algorithmes de simulation		27
	III - 2 - 1. Algorithme de calcul des "données déduites"		29
	III - 2 - 2. Algorithme de calcul des temps d'intervalles initiaux		30
	III - 2 - 3. Algorithme de recherche des lignes concurrentes		31
	III - 2 - 4. Algorithme de traitement des trajets directs		32
	III - 2 - 5. Algorithme de recherche des nœuds de correspondance		33
	III - 2 - 6. Algorithme de calcul de la fonction objectif		34
	III - 2 - 7. Algorithme général		35
IV -	L' O P T I M I S A T I O N		38
	IV - I - Généralités sur l'optimisation		38
	IV - I - 1. Programmation linéaire		41
	IV - I - 2. Programmation non linéaire		43
	IV - 2 - La méthode du gradient		47
V -	C O N C L U S I O N		57

A N N E X E .

## I - INTRODUCTION :

L'objet de cette étude est la mise au point d'un modèle mathématique permettant la simulation d'un réseau de transport en commun urbain et ce, en vue d'une application au calcul du réseau optimal.

Tout d'abord, nous définissons ce que l'on appelle optimisation, ou meilleure façon de faire les choses : c'est l'ensemble des techniques mathématiques permettant de choisir, en fonction de critères définis au préalable, des variables de décision.

Dans tout problème d'optimisation, on retrouve les caractéristiques suivantes : une ou plusieurs fonctions ( fonctions critères ), des variables de décision et des quantités d'informations disponibles pour le choix des variables de décision.

L'essentiel des applications de l'optimisation est de nature économique et les variables de décision sont alors à la disposition d'individus ou de collectivités.

Aussi large que soit son champ d'application, l'optimisation ne doit pas se confondre avec la recherche opérationnelle que l'on définit comme étant l'ensemble des techniques, généralement mathématiques, aidant à la prise de décision.

Ainsi l'objectif de la recherche opérationnelle est de fournir les moyens de prévoir les conséquences de toutes les décisions possibles en utilisant les ressources techniques et scientifiques.

On utilise une représentation du réel chaque fois que cela est plus simple que d'utiliser la réalité elle même. Cela se produit aussi s'il n'est pas possible de manipuler la réalité.



D'une façon générale, il est très rarement avantageux de définir un système optimal par la manipulation d'un système matériel. En effet, la nécessité de faire varier tous les paramètres du système entraînerait la réalisation d'un grand nombre de systèmes distincts. Le coût de l'ensemble de ces réalisations, accru de l'expérimentation de chacune d'elles, serait très élevé.

Le système réel ne pouvant être utilisé, on est amené alors à le remplacer par une représentation appelée modèle, cette opération est appelée simulation. Dans la simulation, il s'agit de remplacer l'objet de l'étude par un modèle plus accessible et manipulable plus aisément.

On distingue deux types de simulations: la simulation mathématique et la simulation physique. On ne s'intéressera, quant à nous, qu'au premier type.

La simulation mathématique est une méthode de description des processus, sous les aspects quantitatif et qualitatif, à l'aide de modèles mathématiques.

Dans la conception d'un modèle mathématique, le processus réel est simplifié, schématisé.

D'un choix convenable du modèle, de la plus ou moins grande précision avec laquelle il traduit les aspects caractéristiques du processus considéré, dépendent la validité de la simulation et la valeur des conclusions obtenues.

Le modèle doit tenir compte de tous les paramètres déterminants du processus, tout en évitant de s'encombrer d'une masse de facteurs peu importants dont la prise en compte ne ferait que rendre les calculs plus complexes.

Le modèle doit permettre une description qualitative et quantitative suffisamment fidèle des caractéristiques et des aspects du processus à étudier.

Combinées aux méthodes de calcul modernes, notamment grâce à la croissance de la capacité des mémoires et de la

rapidité des ordinateurs, les méthodes de simulation permettent, moyennant des dépenses matérielles relativement peu importantes, d'examiner différentes variantes pour le choix des variables de décision.

Ainsi, on est assuré de trouver les solutions optimales dans le cadre du modèle utilisé.

Lors d'une simulation, sur ordinateur, on analyse le processus en faisant varier les divers paramètres dont les liaisons sont exprimées par le modèle mathématique, ce qui permet d'obtenir rapidement des renseignements sur les différentes variantes du processus étudié.

La simulation de certains processus nécessite la prise en compte des contraintes qui existent objectivement sur l'échelle de variations de certains paramètres.

Le modèle mathématique complet d'un processus comprend:

- les variables fondamentales du processus
- les liaisons entre variables fondamentales
- les contraintes sur le processus
- le critère d'optimisation ou fonction objectif

Comme dans un grand nombre de problèmes, un ordinateur est indispensable pour mener à bien les calculs, il va se poser la question de la programmation. La complexité des modèles rend souvent impossible une programmation en langage machine, car elle est trop longue et trop coûteuse; la programmation en un langage très général comme FORTRAN ou ALGOL est très souvent possible mais n'est pas toujours adaptée.

L'analyse d'un grand nombre de modèles de simulation a permis de remarquer qu'on retrouvait souvent les mêmes concepts et qu'il était donc possible de dégager une méthode générale. Il devenait donc nécessaire de concevoir des langages permettant de décrire et d'utiliser commodément les quantités et les relations caractérisant le modèle de simulation de telle sorte que, cette description terminée, il soit possible de le traduire automatiquement pour obtenir un modèle entièrement programmé sur un type d'ordinateur.

C'est pour répondre à cette double question qu'ont été créés les systèmes de simulation. Un système de simulation doit proposer une méthode classique d'analyse des phénomènes exigeant un minimum d'efforts de programmation. A l'heure actuelle, on dénombre plus de cinquante langages de simulation.

Il faut noter deux remarques très importantes:

I - une simulation ne se conçoit pas dans l'absolu, elle se définit uniquement en fonction de l'usage que l'on doit en faire.

II - les solutions trouvées sont celles du modèle et donc ne sont pas d'emblée assimilables à celles du système.

Pour en revenir à l'optimisation, il faut, pour optimiser, définir au préalable la fonction objectif. Le problème de l'optimisation se ramène à trouver l'extrémum de la fonction objectif. Pour cela, différentes méthodes existent, nous les verrons au paragraphe IV.

## II - LA SIMULATION :

### II - I. Présentation du problème :

Les diverses personnes et les divers organismes s'occupant de transports en commun urbains ont des conceptions différentes de ce qu'est un réseau de transport optimal.

L'exploitant d'une société de transport peut choisir comme optimal le réseau qui rapporte le maximum de profits à la société.

Les usagers pensent certainement que le meilleur service est le plus sûr , le plus confortable et le plus rapide.

Les spécialistes en matière de transports urbains estimeront peut être que les installations et services qui peuvent assurer le trafic des passagers sans congestion aux heures de pointes sont les meilleurs.

D'autres penseront peut être que le transport public qui leur offre le service le moins cher est le meilleur.

Ainsi , suivant l'angle sous lequel on voit le problème , la fonction objectif n'est pas la même.

Nous avons , quant à nous, choisi comme fonction objectif le Temps Total de parcours mis par l'ensemble des usagers pour se rendre à leurs destinations respectives , et ce , pendant une tranche horaire donnée.

C'est une grandeur qu'il s'agira de minimiser.

Ainsi , on aura atteint deux objectifs :

- d'une part , la proportion de gens qui se déplacent à pied sera amoindrie
- d'autre part , ceci permettra à la compagnie de transport d'accroître sensiblement ses profits

Vu le but que nous nous sommes assignés , nous avons estimé qu'une simulation était indispensable.

D'une part , en appliquant des règles logiques séquentielles , la simulation permet de reproduire fidèlement le processus.

D'autre part , en réitérant l'évolution du processus un grand nombre de fois , la simulation permet de faire des comparaisons entre un grand nombre de variantes.

Seulement , elle ne permet pas de calculer automatiquement les paramètres pour les fixer à des valeurs optimales pour l'évolution du processus.

## II - 2. Description du processus :

Dans l'étude d'un réseau de transport urbain , interviennent de nombreux facteurs de natures différentes. Ces facteurs sont liés à la compagnie de transport , aux usagers , et au milieu environnant. Certains de ces facteurs sont du type aléatoire.

Dans le processus considéré, un certain nombre d'autobus circulent sur un itinéraire précis. Ils ont pour fonction de véhiculer les usagers qui désireraient se rendre d'un point à un autre du réseau.

Ainsi, dans le processus, interviennent des entités de nature variée : stations, autobus, usagers,....

La description rigoureuse du processus exigerait la prise en considération d'un nombre important de variables et de paramètres caractéristiques du réseau.

Nous avons essayé, dans notre description, de ne considérer que les facteurs déterminants pour l'élaboration du modèle de simulation.

Nous avons ajouté un certain nombre d'hypothèses simplificatrices.

Ainsi, le modèle de simulation doit rendre compte de l'évolution du processus uniquement à partir de ces facteurs.

Nous donnons ici un certain nombre de définitions de termes dont nous parlerons par la suite.

Un réseau de transport urbain se compose d'un certain nombre de lignes desservant un grand nombre de stations.

- noeud :

on appellera noeud du réseau une quelconque de ses stations. Ces noeuds seront repérés par un numéro propre ( généralement  $m, n, o$  ou  $d$  ) totalement indépendant des lignes d'autobus associées.

- station :

une ligne passe par un certain nombre de noeuds qui sont les stations de cette ligne. Les stations sont numérotées en séquence le long de la ligne ( les indices généralement utilisés sont  $i$  ou  $j$  ).

- parcours :

une ligne aller-retour ( comportant deux terminus ) comprend un parcours aller et un parcours retour qui peuvent être différents. De même, une ligne circulaire, ne comportant qu'un seul terminus, peut être constituée d'un parcours dans le sens direct et d'un parcours dans le sens rétrograde. Nous réserverons le mot parcours pour cette utilisation.

Tous les parcours du réseau seront numérotés ( l'indice généralement utilisé sera  $p$  )

Dans la mesure du possible, on respectera la convention suivante : les parcours indicés 21-I et 21 correspondent à la ligne numérotée 1.

- boucle :

sur une ligne aller-retour, les autobus ne décrivent qu'une seule boucle, alors qu'en général, dans le cas d'une ligne circulaire, ils en décrivent deux : la boucle directe et la boucle rétrograde; ces deux dernières peuvent disposer d'un nombre différent d'autobus, avoir des fréquences de passage inégales, et des temps de parcours différents.

On a donc intérêt à différencier réellement ces deux boucles et à considérer dans le modèle les différentes boucles : aller-retour, circulaire directe et circulaire rétrograde. Les boucles seront généralement indicées avec la lettre b.

- ligne :

le mot ligne sera généralement utilisé dans son sens habituel, mais on l'utilisera parfois abusivement dans le sens du mot parcours lorsque la confusion n'est pas possible.

II - 3 - Choix des variables et paramètres :

Le choix des variables et paramètres représentatifs du processus est, comme nous l'avons vu, très important. Il s'agit de sélectionner les facteurs les plus importants.

II - 3 - I - Description des éléments passifs :

Dimensionnement :

- $N_T$  : le réseau comporte  $N_T$  noeuds numérotés de I à  $N_T$  .
- $L_T$  : c'est le nombre de lignes d'autobus .
- $P_T$  : c'est le nombre de parcours différents dans le réseau .
- $B_T$  : nombre de boucles différentes : il est égal au nombre de lignes aller-retour plus deux fois le nombre de lignes circulaires .

Description des itinéraires :

Le parcours indicé  $p$  ( $I \leq p \leq P_T$ ) est caractérisé par les noeuds successivement rencontrés sur le trajet de l'autobus du terminus départ au terminus arrivée.

- $S(p)$  : nombre de stations, terminus inclus.
- $N(p, i)$  : les noeuds successivement rencontrés sur le parcours  $p$  sont donnés pour  $i$  variant de I à  $S(p)$  par le tableau  $N(p, i)$ .
- $I(p, n)$  est le tableau réciproque de  $N(p, i)$ , il fournit le rang du noeud  $n$  sur le parcours  $p$ .

Remarque :

Dans le cas d'un parcours  $p$  d'une ligne circulaire, les terminus de départ et d'arrivée sont les mêmes, on a donc :

$$N(p, I) = N(p, S(p))$$



Dans ce cas, le terminus est la première et la dernière station de la ligne. Par convention, nous prendrons pour le tableau I (p,n) la première de ces déterminations.

III - 3 \_ 2 - Dynamique des éléments actifs :

Dynamique des véhicules

$\theta ( i , j , p ) :$

Le temps de trajet en autobus pour aller de la i-ème station à la j-ème station du parcours p est noté  $\theta ( i , j , p )$ .  $\theta$  n'est défini que pour :

$$I \leq i < j \leq S ( p )$$

$T_R ( b ) :$

Temps de rotation pour les autobus de la boucle b : c'est le temps total qui sépare deux passages consécutifs d'un même autobus par un même arrêt et ce, dans le même sens .

$A_T :$

C'est le nombre d'autobus dont dispose la compagnie de transport, ils ne sont pas forcément tous en circulation sur le réseau .

$X ( b ) :$

temps d'intervalle qui sépare le départ de deux autobus consécutifs sur la boucle b. Si n est le nombre d'autobus en service sur cette boucle, on doit avoir :

$$n.X(b) = T_R ( b )$$

$TI ( p ) :$

temps d'intervalle sur le parcours p. Si le parcours p correspond à la boucle b, alors :

$$TI ( p ) = X ( b )$$

Dynamique des usagers :

- $\tau(m,n)$  : temps de marche à pied pour se rendre d'un noeud  $m$  du réseau à un autre noeud  $n$ . Nous supposons pour simplifier, que  $\tau(m,n) = \tau(n,m)$  ainsi, la matrice  $\tau(m,n)$  sera symétrique.
- $OD(m,n)$  : nombre d'usagers désirant se rendre du noeud  $m$  au noeud  $n$  pendant une certaine tranche horaire.

Comportement des usagers :

- pour se rendre d'un noeud origine "o" à un noeud destination "d", les usagers emprunteront le "plus-court-chemin" en temps moyen de trajet.
- $K(o,d)$  : le premier noeud de correspondance sur ce plus court chemin est  $k_I = K(o,d)$ .

REMARQUES :

- 1 - Le premier noeud de correspondance sur le plus-court-chemin allant de  $k_I$  à  $d$  est  $k_2 = K(k_I, d)$ . C'est donc également le deuxième noeud de correspondance sur le plus-court-chemin de  $o$  à  $d$ , et ainsi de suite.  
De part la définition de la matrice de correspondance  $K$  il n'y a pas de noeud de correspondance entre  $o$  et  $k_I$ .  
Le trajet  $o-k_I$  est donc un trajet direct!
- 2 - Pour tout trajet direct  $m-n$  on aura  $n = K(m,n)$ .  
Le plus court chemin de  $o$  à  $d$  est donc entièrement exploré lorsqu'on trouve pour noeud de correspondance le noeud de destination :

$$k_n = K(k_{n-1}, d) = d.$$

- $I_p(m,n)$  : sur un trajet direct  $m-n$ , on distinguera trois possibilités pour effectuer le trajet, répertoriées dans l'indicateur de possibilités  $I_p(m,n)$ .

$I_p(m,n) = 1$  : tous les usagers effectuent le trajet  $m-n$  en autobus. C'est un "trajet-bus".

$I_p(m,n) = 2$  : le trajet  $m-n$  est effectué à pied par certains et en autobus par d'autres. C'est un "trajet-mixte".

$I_p(m,n) = 3$  : le trajet  $m-n$  est effectué à pied par tous les usagers. C'est un "trajet-pied".

-  $R_A(m,n)$  :  
le pourcentage d'usagers prenant l'autobus sur le trajet direct  $m-n$  est noté  $R_A(m,n)$ .

-  $L(m,n)$  :  
sur le trajet direct  $m-n$ , les usagers peuvent utiliser différentes lignes. Nous appellerons "parcours concurrents" ces lignes. Le nombre de ces parcours concurrents est  $L(m,n)$ .

-  $P(m,n,k)$  :  
les parcours concurrents seront indicés de 1 à  $L(m,n)$  généralement avec la lettre  $k$ . La correspondance entre l'indice  $k$  et le numéro du parcours sera obtenu<sup>e</sup> au moyen du tableau  $P(m,n,k)$ .

-  $R_p(m,n,k)$  :  
la répartition des usagers prenant l'autobus sur le trajet  $m-n$  entre les  $L(m,n)$  parcours concurrents numérotés  $P(m,n,k)$  pour  $k = 1$  à  $L(m,n)$  sera noté  $R_p(m,n,k)$ .

-  $\bar{t}(m,n)$  :  
le temps moyen de parcours du trajet  $m-n$  incluant les temps d'attente, les correspondances, les marches à pied, etc.;... sera noté  $\bar{t}(m,n)$ .

-  $\alpha(m,n,k)$  :  
pour le trajet direct  $m-n$ , la dérivée partielle du temps moyen de parcours  $\bar{t}(m,n)$  par rapport au temps d'intervalle  $TI(p)$  du  $k$ -ième parcours concurrent  $p = P(m,n,k)$  est notée  $\alpha(m,n,k)$ .

$$\alpha(m,n,k) = \frac{d \bar{t}(m,n)}{d TI(p)}$$

REMARQUES :

- 1 - Si  $m-n$  n'est pas un trajet direct,  $I_p(m,n)$ ,  $R_A(m,n)$ ,  $L(m,n)$ ,  $P(m,n,k)$  et  $R_p(m,n,k)$  ne seront pas définis.
- 2 - si  $I_p(m,n) = 1$  ,  $R_A(m,n) = 1$ 
  - si  $I_p(m,n) = 2$  ,  $0 < R_A(m,n) < 1$
  - si  $I_p(m,n) = 3$  ,  $R_A(m,n) = 0$  et  $L(m,n)$ ,  $P(m,n,k)$ ,  $R_p(m,n,k)$  ne seront pas définis.

II - 3 - 3. Variables spécifiques à la collecte et au traitement des données :

Dans un souci de simplicité, nous avons introduit en données les paramètres suivants :

- les tableaux  $S(p)$ ,  $N(p,i)$ ,  $OD(m,n)$ .
- $N_A(m,k)$  : tableau dont l'élément général est le numéro du  $k$ -ième noeud directement adjacent au noeud  $m$ ; deux stations successives sur un parcours d'autobus constituent une paire de noeuds directement adjacents.
- $T_p(m,k)$  : tableau des temps de parcours en autobus entre les noeuds adjacents  $m$  et  $N_A(m,k)$ .
- $\tau(m,n)$  : tableau des temps de parcours à pied; l'élément général de ce tableau est le temps de marche à pied pour se rendre du noeud  $m$  au noeud  $n$ .

Le tableau de rangement des noeuds  $I(p,n)$  est calculé à partir des tableaux  $S(p)$  et  $N(p,i)$ .

Les tableaux des temps de trajet  $\Theta(i,j,p)$  et  $T_A(b)$  seront calculés à partir des tableaux  $S(p)$ ,  $N(p,i)$ ,  $N_A(m,k)$ ,  $T_p(m,k)$ .

II - 4 - Le modèle de simulation :

Le modèle de simulation est l'ensemble des relations permettant de faire évoluer les variables du processus.

En plus des données qu'on fait lire à la machine, nous avons introduit les paramètres suivants :

- $N_T$  : nombre total de noeuds
- $P_T$  : nombre total de parcours
- $A_T$  : nombre total d'autobus
- $N_A^{\max}$  : nombre maximum de noeuds adjacents
- $NL_{\max}$  : nombre maximum de parcours conservés sur un trajet direct
- $NBMPPB$  : capacité des autobus, elle est supposée la même pour tous les autobus.
- $NBMSTA$  : nombre maximum de stations sur un parcours
- $TREPOS$  : temps de repos à un terminus, nous avons supposé qu'il était le même pour tous les parcours.
- $TRANCH$  : *tranche horaire considérée*

II - 4 - I. Données déduites :

Il s'agira, dans une première phase, de calculer les "données déduites", à savoir :  $\theta(i, j, p)$ ,  $Tr(b)$ ,  $I(p, n)$ .

- Calcul de  $I(p, n)$  :  $I(p, n)$  est calculé directement à partir de  $N(p, i)$ .

remarque: si le noeud  $n$  n'appartient pas au parcours  $p$ , alors  $I(p, n) = 0$

- Calcul de  $\theta(i, j, p)$  : sur un parcours donné  $p$ , pour

calculer le temps de trajet en autobus pour aller de la i-ème à la j-ème station, il suffit d'ajouter tous les temps de trajets partiels compris entre i et j, ces temps sont donnés par le tableau  $T_p(m,k)$ .

- Calcul de  $Tr(b)$  :

- pour une ligne aller-retour, le temps de rotation est la somme des temps de trajets pour faire un aller, un retour, plus deux fois le temps de repos à un terminus puisqu'il y a deux terminus.

- pour une ligne circulaire,  $Tr(b)$  est la somme des trajets de trajet en autobus pour décrire la boucle et du temps de repos à un terminus.

II - 4 - 2. Calcul des temps d'intervalle  $X(b)$  ;  $TI(p)$

Ils sont calculés à partir de  $Tr(b)$ .

Le temps d'intervalle qui sépare le départ de deux autobus consécutifs sur la boucle b est égal au rapport du temps de rotation  $Tr(b)$  par le nombre d'autobus affectés sur cette boucle b.

Si on suppose qu'on a le même nombre d'autobus sur les différentes boucles, alors :

$$X(b) = Tr(b) \cdot \frac{B_T}{A_T}$$

Ensuite  $TI(p)$  est déduit de  $X(b)$ .

II - ~~4~~ - 3. Etude des trajets directs :

- Indicateur de possibilités :

Un usager qui désire se rendre à une destination quelconque se présente à la station la plus proche et par laquelle passe une ligne d'autobus vers la destination désirée. Seulement, si l'usager sait qu'il mettrait moins de temps en allant à pied, il éviterait de prendre l'autobus. Il reste cependant un cas à considérer, c'est celui où, suivant que l'usager se présente juste après ou juste avant le départ d'un autobus, il peut décider de partir à pied ou bien préférer attendre le bus, ceci dépend évidemment du temps d'intervalle sur la ligne considérée.

Dans le cas où plusieurs lignes passent par les stations origine et destination considérées, on fait intervenir le temps de trajet minimum et la valeur minimum de l'expression :

$$\theta ( i_m , j_n , p ) + TI ( p )$$

$i_m$  désigne le rang du noeud  $m$  sur le parcours  $p$

$j_n$  " " " " " " " " " " " "

Utilisons l'expression  $\theta ( m, n, p )$  pour désigner  $\theta ( i_m, j_n, p )$

Les tests à faire pour établir l'indicateur de possibilités sont alors les suivants :

$$1 - \tau ( m, n ) \geq \left[ \theta ( m, n, p ) + TI ( p ) \right]_{\min}, \text{ alors } I_p ( m, n ) = 1$$

$$2 - \theta ( m, n, p ) < \tau ( m, n ) < \left[ \theta ( m, n, p ) + TI ( p ) \right]_{\min}$$

$$\text{alors } I_p ( m, n ) = 2$$

$$3 - \theta ( m, n, p ) \geq \tau ( m, n ) \text{ , alors } I_p ( m, n ) = 3$$

- Recherche des parcours concurrents :

Pour un couple de noeuds  $m$  et  $n$ , il suffit de voir combien de lignes passent par ces noeuds. On conserve tous les parcours concurrents si leur nombre ne dépasse pas  $NL_{\max}$ . Sinon, on les range et on ne garde que les  $NL_{\max}$  premiers.

II - 4 - Calcul de la répartition des usagers entre les lignes concurrentes :

Supposons qu'entre deux noeuds quelconques  $m$  et  $n$ , il y ait  $L ( m, n )$  parcours concurrents. Il s'agit de calculer la proportion d'usagers prenant l'autobus sur l'une quelconque de ces lignes.

Considérons l'une quelconque de ces lignes concurrentes ; nous la désignons par l'indice  $l$ , nous allons calculer  $dR_p ( m, n, l )$ , valeur de  $R_p ( m, n, l )$  pendant l'intervalle de temps compris entre  $t$  et  $t+dt$ .

Comme fait, il s'agit d'un calcul de probabilité

Comme en fait il s'agit d'un calcul de probabilité, dRp est le produit des probabilités de deux évènements indépendants "A" et "B" :

- évènement "A" : les usagers qui se présentent entre les instants t et t+dt prennent l'un des autobus de la ligne l si celui-ci se présente seul.
- évènement "B" : aucun des autobus des autres lignes concurrentes ne s'est présenté avant l'instant t.

si on suppose que l'arrivée des usagers à une station est régulière, alors :

$$\text{Prob (A)} = \frac{dt}{\text{TI} (l)}$$

$$\text{Prob (B)} = \prod_{\substack{i=1 \\ i \neq l}}^{L(m,n)} \left[ 1 - \frac{t}{\text{TI} (i)} \right]$$

dRp (m,n,l) est égal à Prob (A).Prob (B)

$$\text{ainsi dRp (m,n,l)} = \frac{1}{\text{TI} (l)} \prod_{\substack{i=1 \\ i \neq l}}^{L(m,n)} \left[ 1 - \frac{t}{\text{TI} (i)} \right] dt$$

Comme il faut calculer Rp (m,n,l) pour toutes les lignes concurrentes sur le trajet m-n, il suffit d'intégrer dRp entre 0 et Tlmin, valeur minimum des temps d'intervalles TI sur les lignes concurrentes.

$$\text{d'où : Rp (m,n,l)} = \int_0^{\text{Tlmin}} \frac{1}{\text{TI} (p_k)} \cdot \prod_{\substack{i=1 \\ i \neq l}}^{L(m,n)} \left( 1 - \frac{t}{\text{TI} (i)} \right) dt$$

L (m,n) désignant le nombre de parcours concurrents sur le trajet m-n et p<sub>k</sub> le parcours concurrent d'indice k.



Calcul de la proportion d'usagers prenant l'autobus sur le trajet direct m-n :

$R_A(m,n)$  est la proportions d'usagers prenant l'autobus sur le trajet direct m-n.

$I - R_A(m,n)$  désigne alors la proportions de gens effectuant ce trajet à pied.

Soit  $z$  le temps d'attente,  $\tau$  le temps de trajet à pied et  $\theta$  le temps de trajet en autobus.

$R_A(m,n)$  peut s'écrire :

$$R_A(m,n) = \text{Prob}(\tau > z + \theta) = \text{Prob}(z < \tau - \theta)$$

d'où l'on déduit :

$$R_A(m,n) = F_R(\tau - \theta)$$

Nous verrons dans le calcul de la fonction de répartition du temps d'attente que  $F_R(t)$  a l'expression suivante :

$$F_R(t) = I - \prod_{i=I}^{L(m,n)} \left[ I - \frac{t}{T_i(i)} \right]$$

l'expression de  $R_A(m,n)$  s'en déduit aisément :

$$R_A(m,n) = I - \prod_{i=I}^{L(m,n)} \left[ I - \frac{\tau - \theta}{T_i(i)} \right]$$

II - 4 - 4. Fonction de répartition du temps d'attente :

Soit  $F_R(t)$  la fonction de répartition du temps d'attente.  
Si  $z$  désigne le temps d'attente,  $F_R(t)$  s'écrit alors :

$$F_R(t) = \text{Prob}(z < t) = I - \text{Prob}(z > t)$$

on peut écrire  $\text{Prob}(z > t)$  sous la forme suivante :

$$\text{Prob}(z > t) = \prod_{i=I}^{L(m,n)} \left[ \text{Prob}_i(z > t) \right]$$

$$\text{Prob}(z > t) = \prod_{i=I}^{L(m,n)} \left[ I - \text{Prob}_i(z < t) \right]$$

$$\text{comme } \text{Prob}_i(z < t) = \frac{t}{\text{TI}(i)}$$

$$\text{alors } \text{Prob}(z > t) = \prod_{i=I}^{L(m,n)} \left[ I - \frac{t}{\text{TI}(i)} \right]$$

$$F_R(t) \text{ s'écrit donc : } F_R(t) = I - \prod_{i=I}^{L(m,n)} \left[ I - \frac{t}{\text{TI}(i)} \right]$$

Temps d'attente moyen :

-  $I_p(m,n) = I$  :

Soit  $E(z)$  le temps d'attente moyen.  $F_R(t)$  étant la fonction de répartition du temps d'attente, on a, par définition:

$$E(z) = \int_{-\infty}^{+\infty} t \cdot dF_R(t)$$

$$\begin{aligned} F_R(t) &= 0 \quad \text{pour } t < 0 \\ &= F_R \quad t < T_{\text{Imin}} \\ &= I \quad t > T_{\text{Imin}} \end{aligned}$$

$T_{\text{Imin}}$  est le temps d'intervalle minimum sur les différents parcours concurrents.

$E(z)$  s'écrit alors :

$$E(z) = \int_0^{T_{\text{Imin}}} t \cdot dF_R(t)$$

$2 - I_p(m,n) = 2$  :

$$E(z) = \int_{-\infty}^{+\infty} t \cdot dF_R(t)$$

$$\begin{aligned} F_R(t) &= 0 \quad \text{pour } t < 0 \\ &= F_R \quad t < \lambda - \theta \\ &= c^{te} \quad t > \lambda - \theta \end{aligned}$$

$E(z)$  a donc pour expression :

$$E(z) = \int_0^{\lambda - \theta} t \cdot dF_R(t)$$

Reprenons maintenant les expressions de  $F_R(t)$ ,  $R_A(m,n)$  et  $R_p(m,n,k)$ .

$$F_R(t) = I - \prod_{i=I}^{L(m,n)} \left[ I - \frac{t}{TI(i)} \right]$$

$$R_A(m,n) = I - \prod_{i=I}^{L(m,n)} \left[ I - \frac{\tau - \theta}{TI(i)} \right]$$

$$R_p(m,n,k) = \int_0^{TI_{min}} \frac{I}{TI(p_k)} \prod_{\substack{i=I \\ i \neq k}}^{L(m,n)} \left[ I - \frac{t}{TI(i)} \right] dt$$

Nous remarquons que ces trois expressions présentent un aspect commun. Au lieu d'avoir à calculer à chaque séquence ces expressions, il est plus simple d'écrire un sous-programme qu'on appellera chaque fois que cela est nécessaire. Pour cela il faut écrire ces expressions sous une forme plus pratique:

considérons l'expression de  $F_R(t)$  :

$$F_R(t) = I - \prod_{i=I}^{L(m,n)} \left[ I - \frac{t}{TI(i)} \right]$$

Supposons que, sur un trajet direct  $m-n$ , il y ait trois parcours concurrents indicés  $p_1$ ,  $p_2$ ,  $p_3$ ; alors :

$$F_R(t) = I - \prod_{i=I}^3 \left[ I - \frac{t}{TI(i)} \right]$$

$$F_R(t) = I \left[ I - \frac{t}{TI(p_1)} \right] \cdot \left[ I - \frac{t}{TI(p_2)} \right] \cdot \left[ I - \frac{t}{TI(p_3)} \right]$$

$$F_R(t) = t \cdot \left[ \frac{I}{TI(p_1)} + \frac{I}{TI(p_2)} + \frac{I}{TI(p_3)} \right]$$

$$- t^2 \cdot \left[ \frac{I}{TI(p_2) \cdot TI(p_3)} + \frac{I}{TI(p_1) \cdot TI(p_2)} + \frac{I}{TI(p_1) \cdot TI(p_3)} \right]$$

$$+ t^3 \cdot \left[ \frac{I}{TI(p_1) \cdot TI(p_2) \cdot TI(p_3)} \right]$$

Nous voyons que  $F_R(t)$  peut s'exprimer sous une forme polynomiale, il suffit d'écrire un sous programme du type FUNCTION qui calculera les coefficients du polynome.

Ainsi  $F_R(t)$  aura l'expression suivante :

$$F_R(t) = - \sum_{r=1}^{L(m,n)} E_{L(m,n)}^r(a) (-t)^r$$

$L(m,n)$  est le nombre de parcours concurrents sur  $m-n$ .

$$a_k = \frac{I}{TI(p_k)}$$

De même,  $R_A(m,n)$  s'écrira :

$$R_A(m,n) = - \sum_{r=1}^{L(m,n)} E_{L(m,n)}^r(a) (\theta - \tau)^r$$

$$R_p(m, n, k) = \frac{TI_{min}}{TI(p_k)} \sum_{r=0}^{L(m, n) - I} \frac{I}{r + I} \cdot E(a_k) \cdot (-TI_{min})^r$$

Ainsi, on peut écrire l'expression du temps moyen d'attente sous la forme polynomiale :

- Ip (m, n) = I .

$$E(z) = \int_0^{TI_{min}} t \cdot dF_R(t) = W(TI_{min})$$

- Ip (m, n) = 2 .

$$E(z) = \int_0^{\lambda - Y} t \cdot dF_R(t) = W(\lambda - Y)$$

W(t) ayant pour expression :

$$W(t) = \sum_{r=I}^{L(m, n)} \left( \frac{r}{r+I} \right) \cdot E(a_k) \cdot (-t)^{r+I}$$

II - 4 - 5). Temps de trajet moyen sur chaque trajet direct :

Soit Y le temps moyen mis par l'ensemble des usagers qui se déplacent du noeud m au noeud n en empruntant les différents parcours concurrents.

$$Y = \sum_{k=I}^{L(m, n)} R_p(m, n, k) \cdot \theta(i_m, j_n, k)$$

Différents cas peuvent se produire :

I .  $I_p(m,n) = 1$

$$\bar{t}(m,n) = E(z) + Y$$

2 .  $I_p(m,n) = 2$

$$\bar{t}(m,n) = E(z) + R_A(m,n)Y + (1 - R_A(m,n)) \cdot \zeta(m,n)$$

3 .  $I_p(m,n) = 3$

$$\bar{t}(m,n) = \zeta(m,n)$$

II - 4 - ①. Etude des trajets non directs :

Recherche des noeuds de correspondance :

Etant donné deux noeuds quelconques o et d, il s'agit, dans le cas où o et d n'est pas un trajet direct, de chercher tous les noeuds de correspondance entre o et d.

II - 4 - ②. Calcul de la fonction objectif :

La fonction objectif est le temps de trajet moyen mis par l'ensemble des usagers pour aller à leurs destinations respectives pendant la tranche horaire considérée. Si TTPA désigne la fonction objectif :

$$TTPA = \sum_{\substack{o=1 \\ d=1}}^{N_T} OD(o,d) \cdot \bar{t}(o,d)$$

II - 4 - 8. Calcul du nombre de passagers par trajet direct :

OD (m,n) étant la demande origine-destination

$R_A$  (m,n) la proportion de gens qui prennent l'autobus sur la ligne considérée

Rp (m,n,k) le pourcentage d'usagers sur la ligne concurrente de rang k

le nombre de passagers sur la ligne k , pour le trajet direct m-n, est :

$$OD (m,n) \cdot R_A (m,n) \cdot Rp (m,n,k)$$

le nombre de passagers sur le trajet direct m-n s'en déduit aisément :

$$PAS (m,n) = \sum_{k=1}^{L(m,n)} OD (m,n) \cdot R_A (m,n) \cdot Rp (m,n,k)$$

II - 4 - 9. Calcul des chargements à chaque inter-station :

Partant d'un couple de noeuds (m,n), on fait intervenir

$I_p$  (m,n).

a.  $I_p$  (m,n) = 1 :  $PAS (m,n) = \sum_{k=1}^{L(m,n)} OD (m,n) \cdot Rp (m,n,k)$

b.  $I_p$  (m,n) = 2 :  $PAS(m,n) = \sum_{k=1}^{L(m,n)} OD(m,n) \cdot R_A (m,n) \cdot Rp(m,n,k)$

c.  $I_p$  (m,n) = 3 :

1. s'il n'y a aucun noeud de correspondance entre m et n, alors  $\bar{t} (m,n) = \zeta (m,n)$  et  $PAS (m,n) = 0$

2. il y a un noeud de correspondance entre m et n soit  $N_d = K (m,n)$ ,  $\bar{t} (m,n) = \bar{t} (m,N_d) + \bar{t} (N_d,n)$



$$PAS = \sum_{k=I}^{L(m,n)} OD(m,n) \cdot R_A(m,n) \cdot Rp(m,n,k)$$

Le nombre de passagers étant déterminé sur chaque trajet m-n, on calcule pour chaque boucle les chargements aux inter-stations en tenant compte des trajets avec noeuds de correspondance et on fait la somme des chargements pour chaque boucle. On détermine ainsi pour chaque boucle le nombre de passagers qui prendraient l'autobus. Soit PASMAL (NL) ce nombre pour la boucle indiquée NL, alors le temps d'intervalle maximum sur la boucle NL est :

$$TIMAX (NL) = TRANCH. \frac{NBMPPB}{PASMAL (NL)}$$

II - 4 - 10. Calcul du nombre d'autobus nécessaire :

Pour calculer le nombre d'autobus nécessaire pour véhiculer l'ensemble des usagers , il faut auparavant déterminer le temps d'intervalle maximum sur chaque boucle.

Si NEMPPB désigne la capacité de chaque autobus  
PASMAL (NL) le chargement maximum sur la boucle NL  
TRANCH la tranche horaire considérée

$$\text{alors : } \text{TIMAX (NL)} = \text{TRANCH} \cdot \frac{\text{NEMPPB}}{\text{PASMAL (NL)}}$$

Ainsi la valeur de NBNEC, le nombre d'autobus nécessaire, est:

$$\text{NBNEC} = \sum_{\text{NL}=1}^{\text{B}_T} \frac{\text{T}_R (\text{NL})}{\text{TIMAX (NL)}}$$

Ce nombre sera comparé par la suite au nombre de véhicules dont dispose la compagnie.

### III - LA PROGRAMMATION :

#### III- I. LE programme de simulation :

L'organigramme général du programme de simulation représente les liaisons qui existent entre les différents sous-programmes. Le programme principal est en relation avec six sous-programmes du type SUBROUTINE et un sous-programme du type FUNCTION.

Les sous-programmes utilisés sont généraux, utilisables tels quels sans aucune modification quelque soit le réseau à simuler.

On fait appel à ces sous-programmes au moyen du programme principal dans lequel on fait lire les paramètres et données particuliers au réseau considéré.

Le programme doit pouvoir être utilisé pour simuler le réseau pour n'importe quelle tranche horaire. Il suffit pour cela de faire lire la tranche horaire considérée et modifier la matrice de demande origine-destination OD (m,n).

L'algorithme utilisé est programmé en FORTRAN IV, ce langage général se prête bien au problème considéré.

Le programme est très général, il doit permettre la simulation de n'importe quel réseau de transport urbain.

Ce programme doit calculer principalement, pour un réseau donné, la valeur de la fonction objectif.

Nous présentons en annexe un exemple d'application du programme que nous proposons.

#### III- 2. Principaux algorithmes de simulation :

Le programme de simulation comporte sept algorithmes :

I - Un algorithme de calcul des "données déduites", auquel correspond le sous-programme PARAM, dont l'algorithme est représenté à la figure I.

2 - Un algorithme de calcul des temps d'intervalles initiaux, auquel correspond le sous-programme TIINI, dont l'algorithme est représenté à la figure II.

3 - Un algorithme de recherche des lignes concurrentes, auquel correspond le sous-programme LIGNES, dont l'organigramme est représenté à la figure III.

4 - Un algorithme traitant les trajets directs, auquel correspond le sous-programme TRAMOY, dont on a représenté l'organigramme à la figure IV.

5 - Un algorithme de recherche des noeuds de correspondance auquel on fait correspondre le sous-programme COREPO, dont l'organigramme est représenté à la figure V.

6 - Un algorithme de calcul de la fonction objectif, auquel on fait correspondre le sous-programme AFFECT, dont l'organigramme est représenté à la figure VI.

7 - Un algorithme général assurant la coordination de ces six algorithmes, son organigramme est représenté à la figure VII.

1

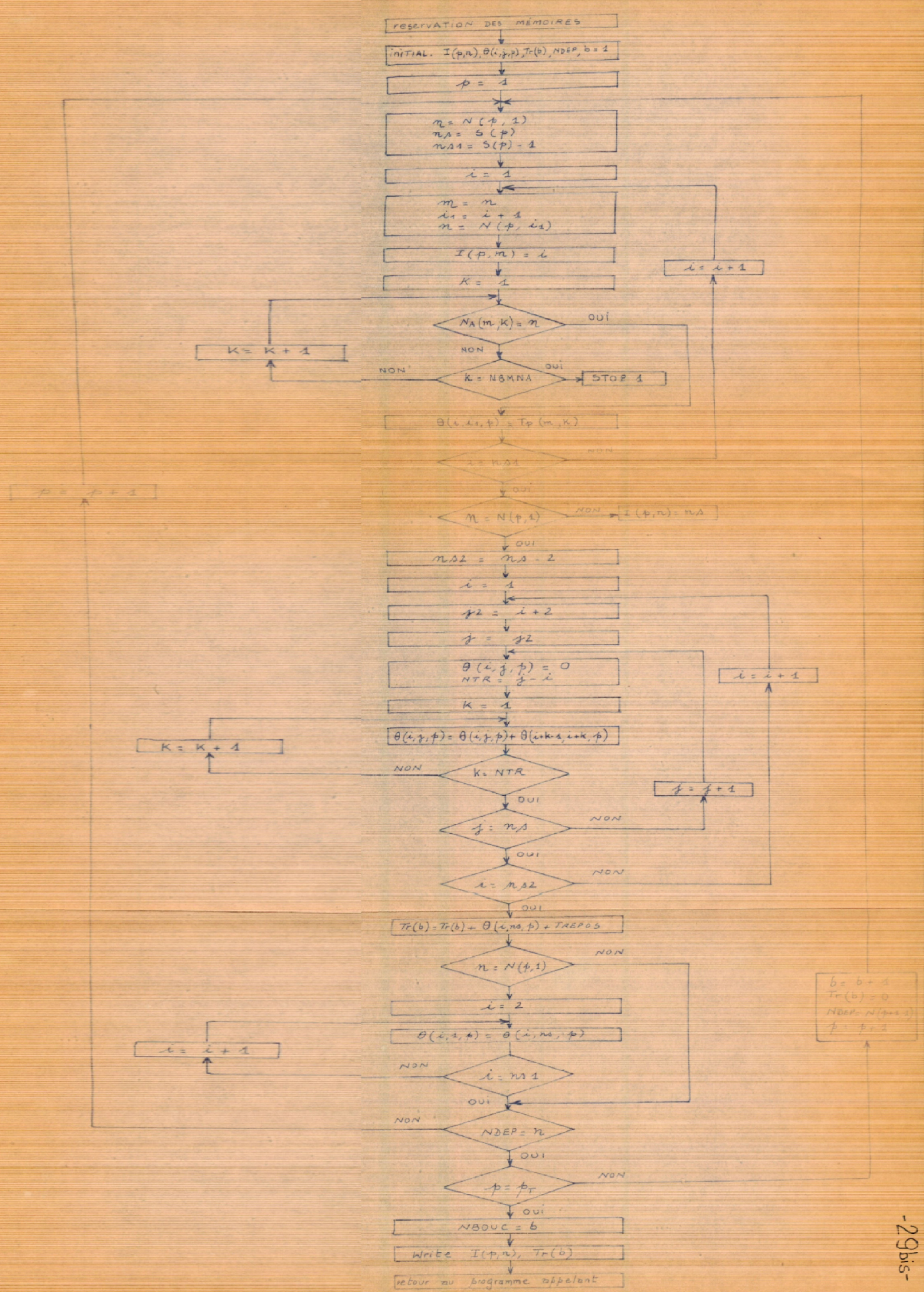
Nous donnons à la figure VIII l'organigramme de calcul des coefficients du polynôme, correspondant au sous-programme du type FUNCTION, que nous avons appelé COFPOL.

III - 2 - I. Algorithme de calcul des "dangées déduites" :

A partir du tableau  $T_p(m,k)$ , cet algorithme calcule les temps de trajets en autobus entre deux stations consécutives, entre deux stations quelconques sur un parcours, et permet de déduire les temps de rotation  $Tr(b)$  sur chacune des boucles du réseau.

En même temps, étant donné un noeud quelconque, cet algorithme fournit le rang de ce noeud sur chacun des parcours. Si le noeud  $n$  ne se trouve pas sur le parcours  $p$ ; alors  $I(p,n) = 0$ .

FIG. 1 ORGANIGRAMME DE L'ALGORITHME N° 1



III - 2 - 2. Algorithme de calcul des temps d'intervalles initiaux :

Le sous-programme TIINI, correspondant à cet algorithme, permet de calculer les temps d'intervalles  $TI(p)$  pour tous les parcours du réseau.

Les lignes aller-retour sont numérotées en premier, à chaque ligne  $l$ , correspond une boucle  $b=l$  et deux parcours  $p$  et  $p+I$  tels que :

$$p = 2l - I$$

Les lignes circulaires sont numérotées ensuite. Le premier parcours correspondant sera trouvé lorsque la relation  $I(p, N(p,I)) = I$  est vérifiée.

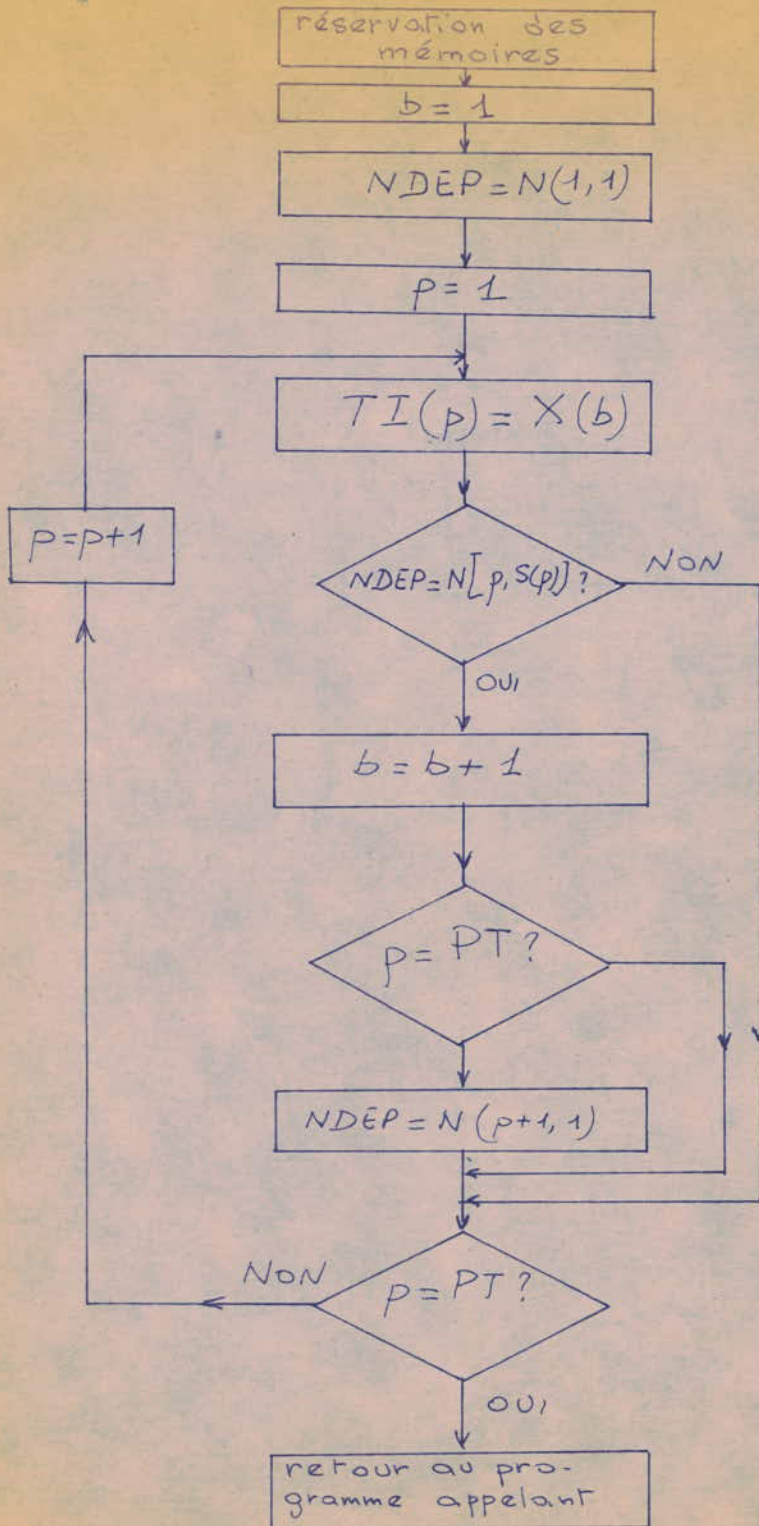


Fig II - ORGANIGRAMME DE L'ALGORITHME N° 2



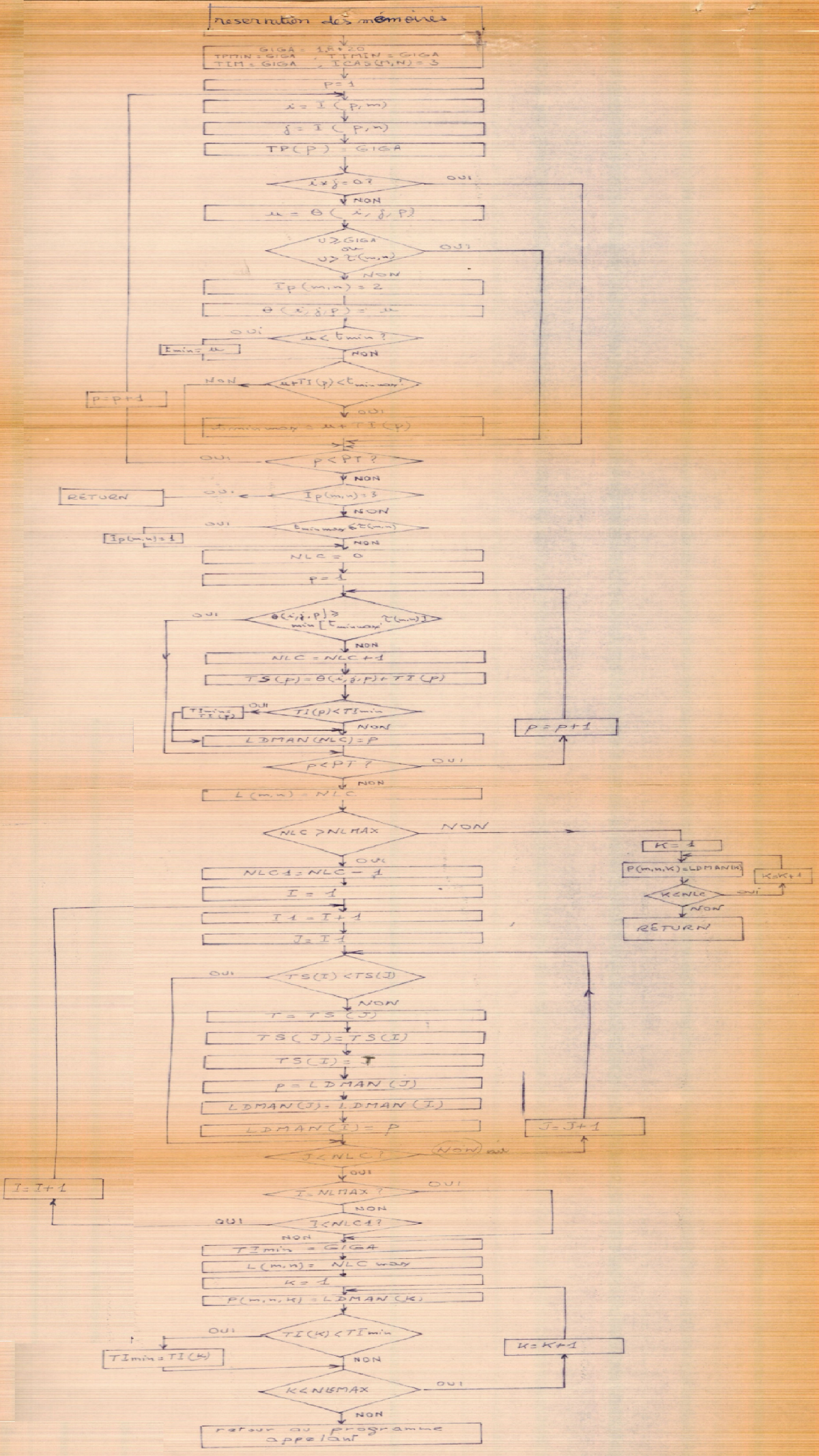


Fig III ORGANIGRAMME DE L'ALGORITHME N°3.

III - 2 - 3. Algorithme de recherche des lignes concurrentes:

Pour un couple de noeuds  $(m,n)$ , il s'agit de voir s'il passe au moins une ligne d'autobus par  $m$  et  $n$ , dans le sens  $m$  vers  $n$ , donc de voir si le trajet  $m-n$  est un trajet direct.

Ensuite, pour tout trajet direct, il faut déterminer l'indicateur de possibilités, connaissant le temps de marche à pied pour aller de  $m$  à  $n$ , le temps de trajet en autobus et le ou les temps d'intervalles suivant qu'une ou plusieurs lignes passent par  $m$  et  $n$ .

Comme il peut passer plusieurs lignes par les noeuds origine et destination considérés, il faudra déterminer le nombre  $L(m,n)$  et les numéros de ces lignes.

On s'est fixé au départ le nombre maximum de parcours concurrents  $NLMAX$  à conserver sur tout trajet direct, il est possible que  $L(m,n)$  soit plus grand que  $NLMAX$ , dans ce cas, il faut ranger les parcours concurrents par ordre "croissant" en fonction des valeurs des expressions de

$$TS(p) = \Theta(i_m, j_n, p) + TI(p)$$

ensuite, il suffira de ne conserver que les  $NLMAX$  premiers.

Les parcours concurrents sont conservés dans le tableau  $P(m, n, k)$ .

III - 2 - 4. Algorithme traitant les trajets directs :

L'objet de cet algorithme est de recenser tous les trajets directs du réseau , pour cela, il fait appel au départ au sous-programme de l'algorithme de recherche des lignes concurrentes. C'est donc un algorithme qui ne traite que les trajets directs. Si, pour un trajet m-n,  $I_p(m,n) = 3$ , alors :

$$\bar{v}(m,n) = \bar{z}(m,n)$$

si  $I_p(m,n) = 1$  ou  $2$ , cet algorithme calcule les valeurs de  $R_p(m,n,k)$  et  $R_A(m,n)$  en utilisant leurs expressions polynomiales.

Le sous-programme correspondant à cet algorithme fait donc appel au sous-programme de calcul des coefficients des polynomes COFPOL.

Eventuellement, on pourra imprimer les tableaux  $I_p(m,n)$ ,  $L(m,n)$ ,  $P(m,n,k)$ ,  $R_p(m,n,k)$  et  $R_A(m,n)$ .

III - 2 - 5. Algorithme de recherche des noeuds de correspondance :

L'objet de cet algorithme est de déterminer tous les noeuds de correspondance entre un noeud origine et un noeud destination lorsque aucune ligne ne les relie ou bien lorsque le trajet considéré est un "trajet-pied", il s'agira d'essayer de l'améliorer.

Cependant, on procédera de la façon suivante : on considèrera chaque noeud et on regarde pour quel couple de noeuds celui-ci est premier noeud de correspondance.

Si pour le couple de noeuds m et n, k est le premier noeud de correspondance, alors :

$$K(m,n) = K(m,k) = k$$

$$\text{et } \bar{t}(m,n) = \bar{t}(m,k) + \bar{t}(k,n)$$

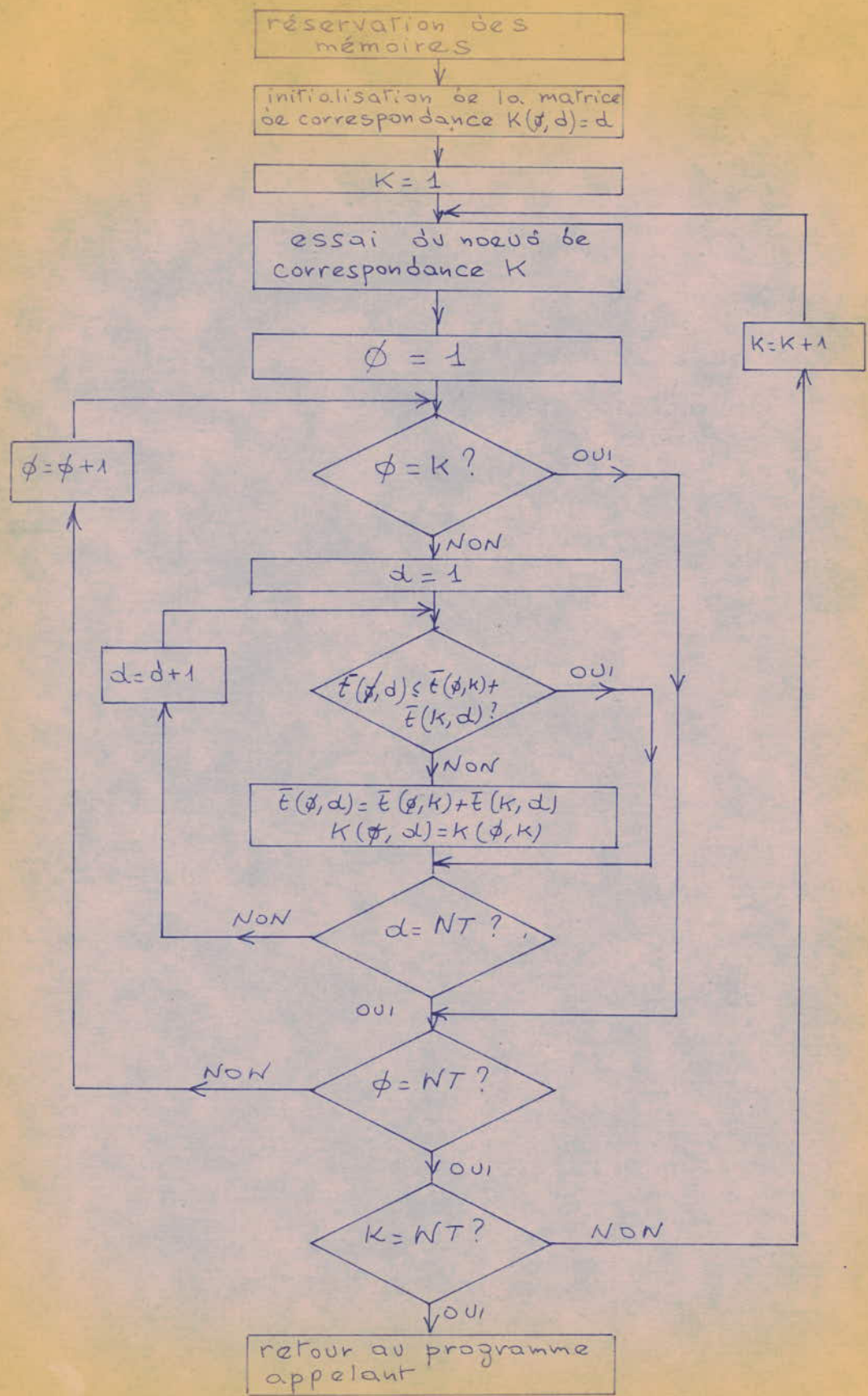


Fig V. ORGANIGRAMME DE L'ALGORITHME N°5.

III - 2 - 6. Algorithme de calcul de la fonction objectif :

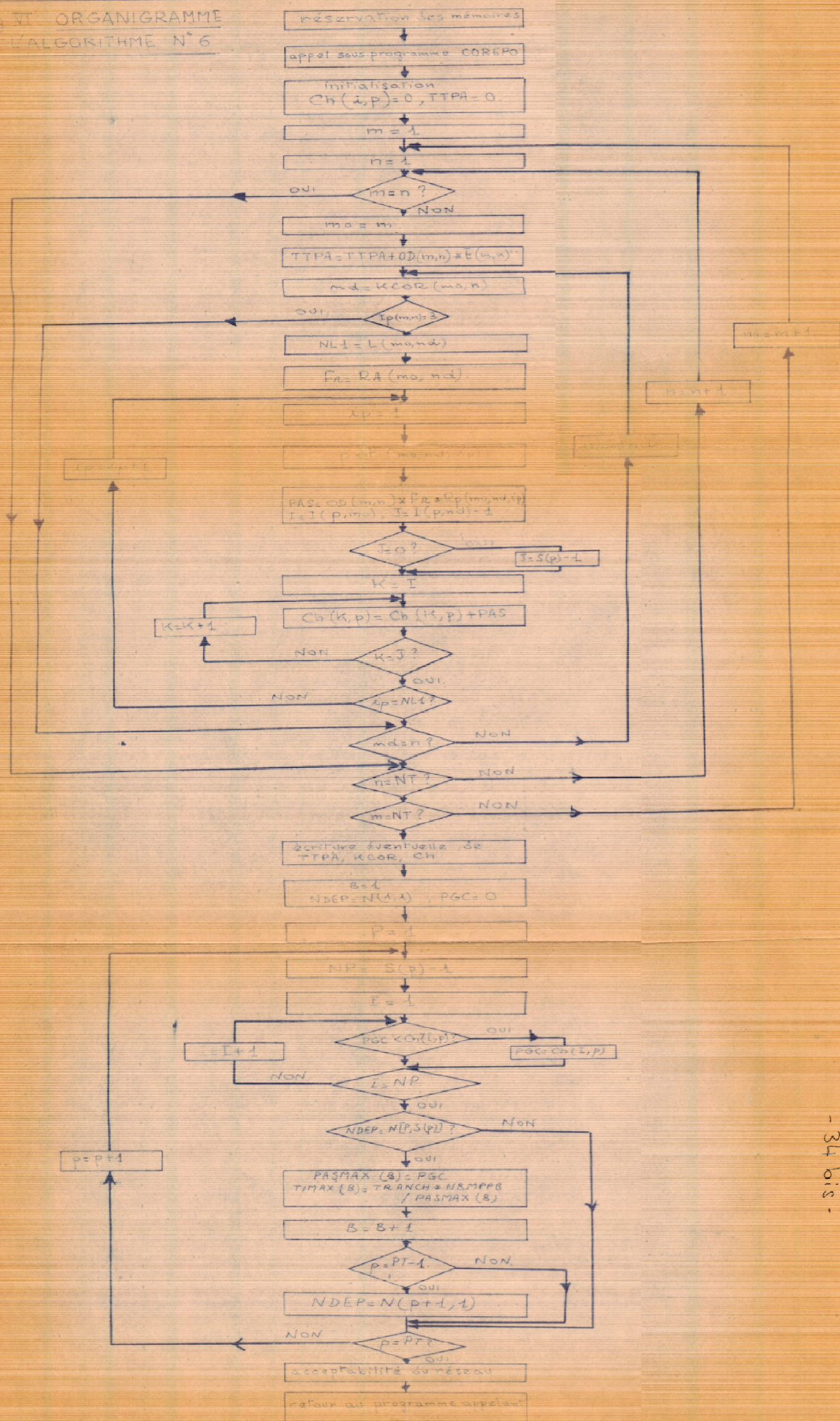
Une fois déterminés tous les noeuds de correspondance éventuels pour chaque trajet, il s'agit, connaissant tous les temps de trajet moyens et la demande origine-destination, de trouver la valeur de la fonction objectif, d'après la relation:

$$TTPA = \sum_{\substack{m=1 \\ d=1}}^{N_T} OD(m,n) \cdot \bar{t}(m,n)$$

Cet algorithme calcule aussi les chargements à chaque interstation, les temps d'intervalles maximum sur chaque boucle, ainsi que le nombre d'autobus nécessaire, et ce, en utilisant les relations données dans le modèle.

Finalement, on a un test d'acceptabilité du réseau. Si le nombre d'autobus calculé est plus grand que celui dont dispose la compagnie, il faut modifier l'affectation des autobus sur les différentes boucles. Sinon, on dira que le réseau peut être acceptable.

Fig VI ORGANIGRAMME DE L'ALGORITHME N° 6



III - 2 - 7. Algorithme général :

Nous avons supposé au début que la matrice des temps de marche à pied était symétrique. Nous calculerons donc la matrice triangulaire inférieure puisque nous avons introduit la matrice triangulaire supérieure. L'algorithme général est relié aux algorithmes de calcul des données déduites, de calcul des temps d'intervalles initiaux, de traitement des trajets moyens et de calcul de la fonction objectif.

Le calcul du nombre d'autobus utilisé est effectué dans cet algorithme. Si on décide d'utiliser 80% du nombre de véhicules dont dispose la société de transport, alors :

$$NBUT = 0,8 \cdot NBTBUS$$

Si on suppose que le nombre d'autobus est le même sur toutes les boucles, alors chaque boucle disposera de:

$$NBBUS = \frac{NBUT}{B_T}$$



- 35 bis -

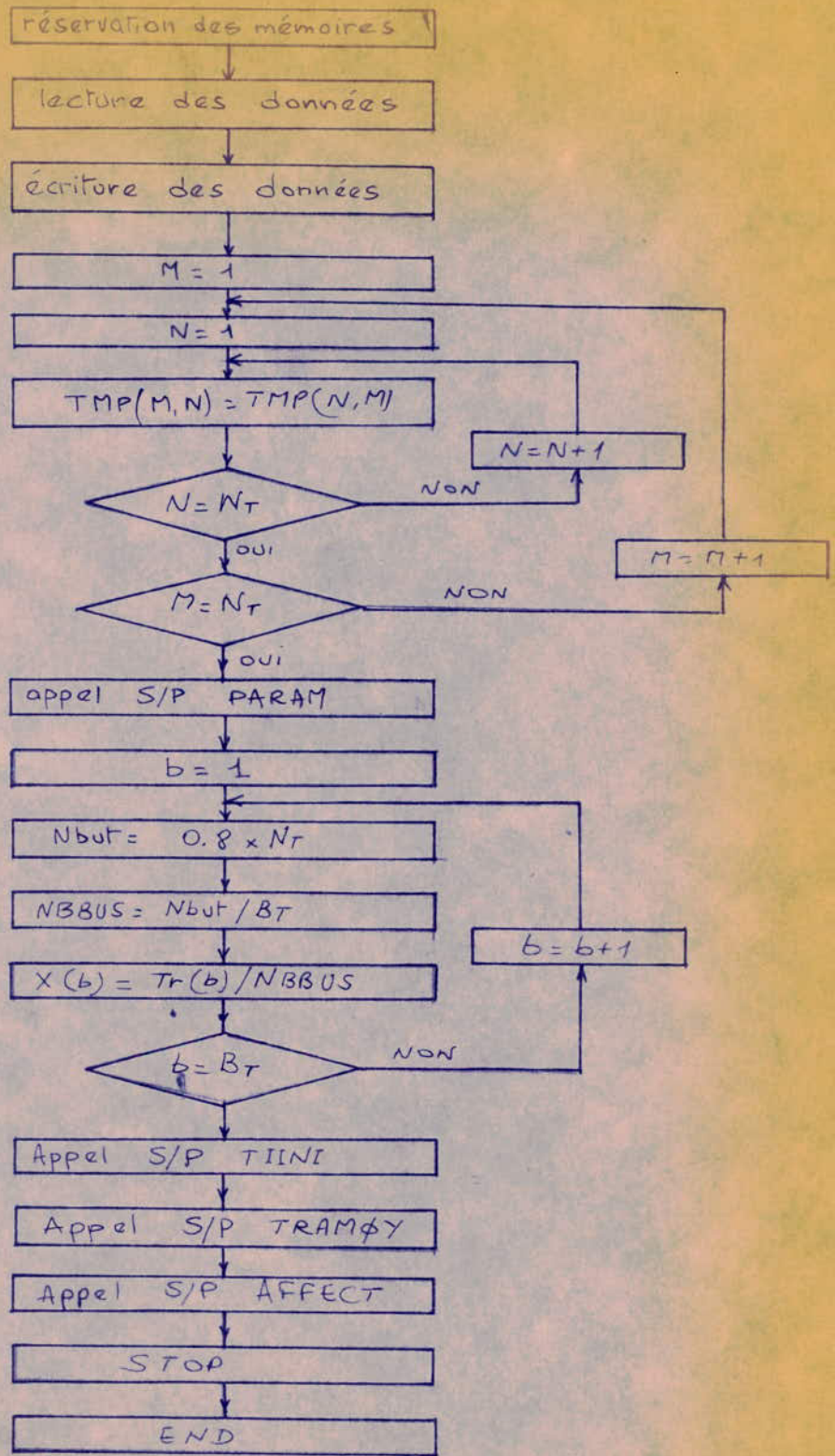
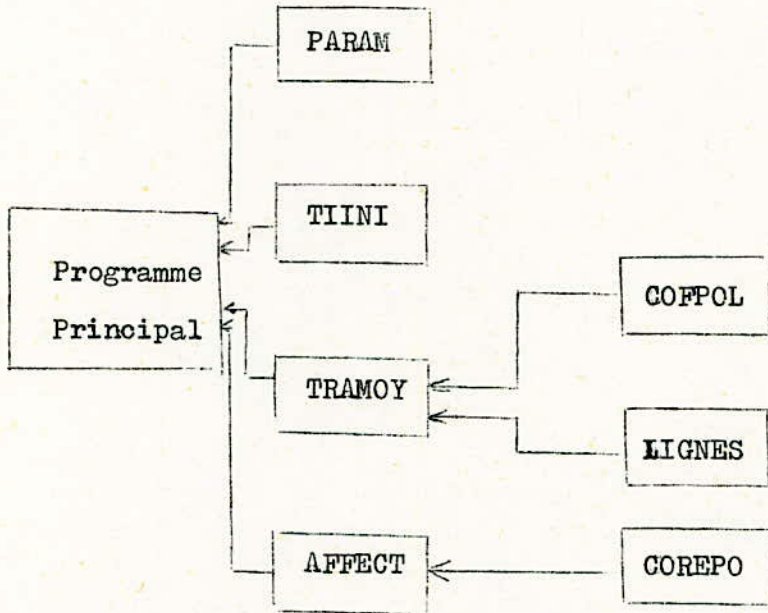


Fig VII - ORGANIGRAMME DE L'ALGORITHME GENERAL



Liaisons entre les différents sous-programmes.

Variables du programme :

Entre crochets nous mentionnons les noms des variables dans le modèle.

NBTNEU	:	$\left[ N_T \right]$
NBTPAR	:	$\left[ P_T \right]$
NBOUC	:	$\left[ B_T \right]$
NEMSTA	:	nombre maximum de stations sur un parcours
NLMAX	:	nombre maximum de parcours concurrents conservés sur les trajets directs
NEMNA	:	nombre maximum de noeuds adjacents à un autre noeud
NBTBUS	:	$\left[ A_T \right]$
NEMPPB	:	nombre maximum de passagers par autobus, c'est aussi la capacité d'un autobus
NBSTA (L)	:	$\left[ S(p) \right]$
NEULI (I,L)	:	$\left[ N(p,i) \right]$
NOSTA (N,L)	:	$\left[ I(p,n) \right]$
NONA (M,K)	:	$\left[ N_A(m,k) \right]$
TPBIN (M,K)	:	$\left[ T_p(m,k) \right]$
TPBUS (I,J,L)	:	$\left[ \theta(i,j,p) \right]$
TPTT (NL)	:	$\left[ Tr(b) \right]$
TI (L)	:	$\left[ TI(p) \right]$
X (NL)	:	$\left[ X(b) \right]$
TMP (M,N)	:	$\left[ \tau(m,n) \right]$
TTM (M,N)	:	$\left[ \bar{t}(m,n) \right]$
ICAS (M,N)	:	$\left[ I_p(m,n) \right]$
NLMN (M,N)	:	$\left[ L(m,n) \right]$
LMN (M,N,K)	:	$\left[ P(m,n,k) \right]$

PROB (M,N,K) :  $[R_p(m,n,k)]$   
PBUS (M,N) :  $[R_A(m,n)]$   
IOD (M,N) :  $[OD(m,n)]$   
KCOR (M,N) :  $[K(m,n)]$   
NEPPST (I,L) : Ch (i,p) : chargement à l'inter-station i  
sur le parcours p.  
PASMAY (NL) : chargement maximum sur la boucle b.  
TIMAX (NL) : temps d'intervalle maximum sur la boucle b.  
TTPA : temps total de parcours, c'est la fonction  
objectif.  
TREPOS : temps de repos à un terminus  
TRANCH : tranche horaire considérée  
~~NEBUS~~  
NEBEC : nombre d'autobus nécessaire.

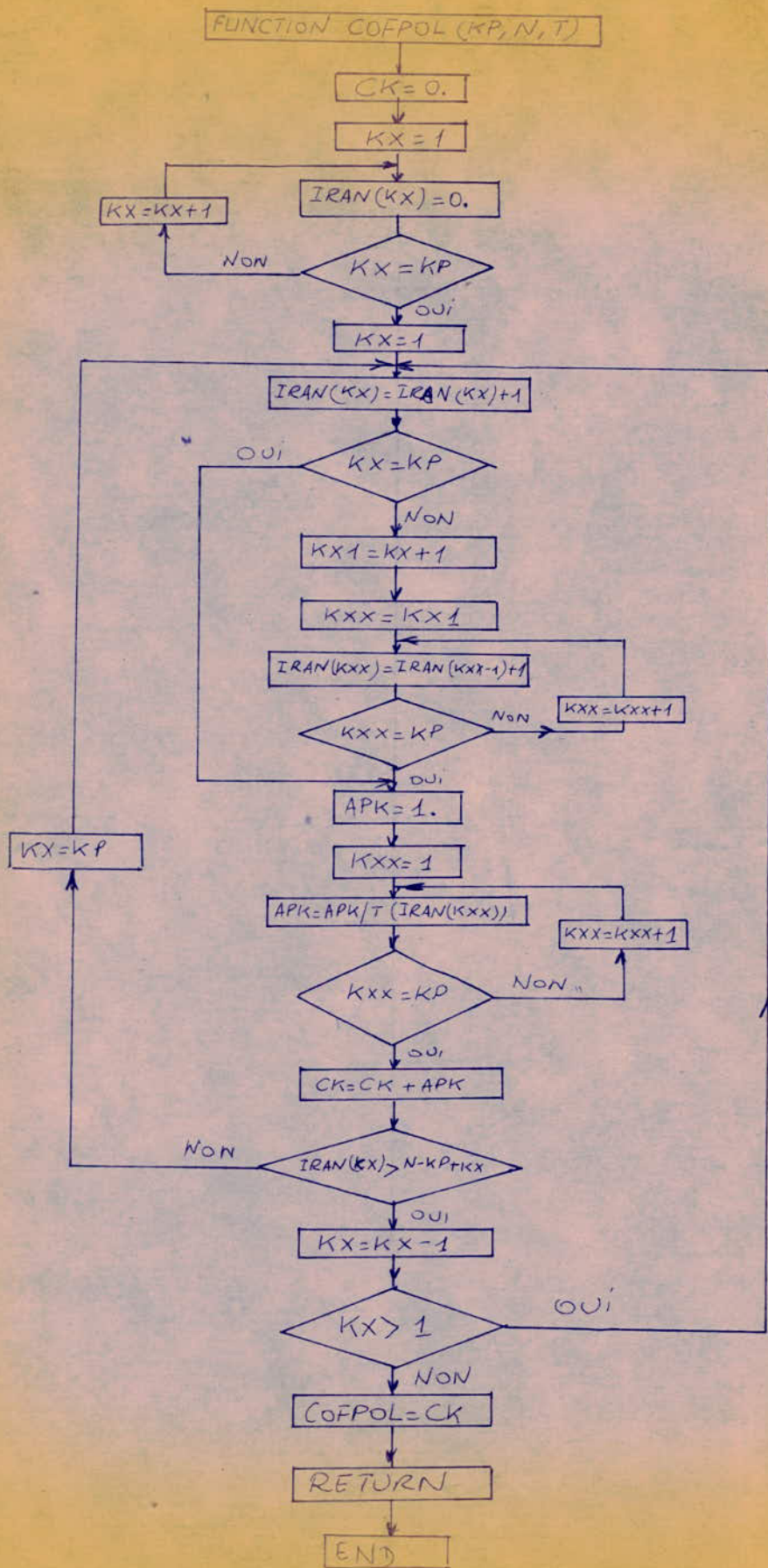


Fig VIII - Organigramme du sous-programme COFPOL.

IV - L'OPTIMISATION

---

IV - I - Généralités sur l'optimisation :

---

Les problèmes d'optimisation sont infiniment variés . Ils intéressent non seulement les ingénieurs, les physiciens, les statisticiens, mais aussi tous ceux qui s'occupent de mathématiques appliquées, d'économie, d'analyse opérationnelle et d'études de systèmes .

Pendant deux siècles, on ne disposa pour résoudre les problèmes d'optimisation, que des méthodes classiques du calcul différentiel et du calcul de variation .

Avec l'apparition de la recherche opérationnelle, on assista à un regain d'intérêt pour les méthodes d'optimisation permettant de résoudre des problèmes non solubles par les méthodes classiques ; la programmation linéaire et la programmation dynamique sont des techniques modernes de recherche d'un optimum .

Nous donnons ici quelques définitions de termes utilisés dans la recherche d'un optimum . Nous nous servons pour cela d'un exemple relativement simple .

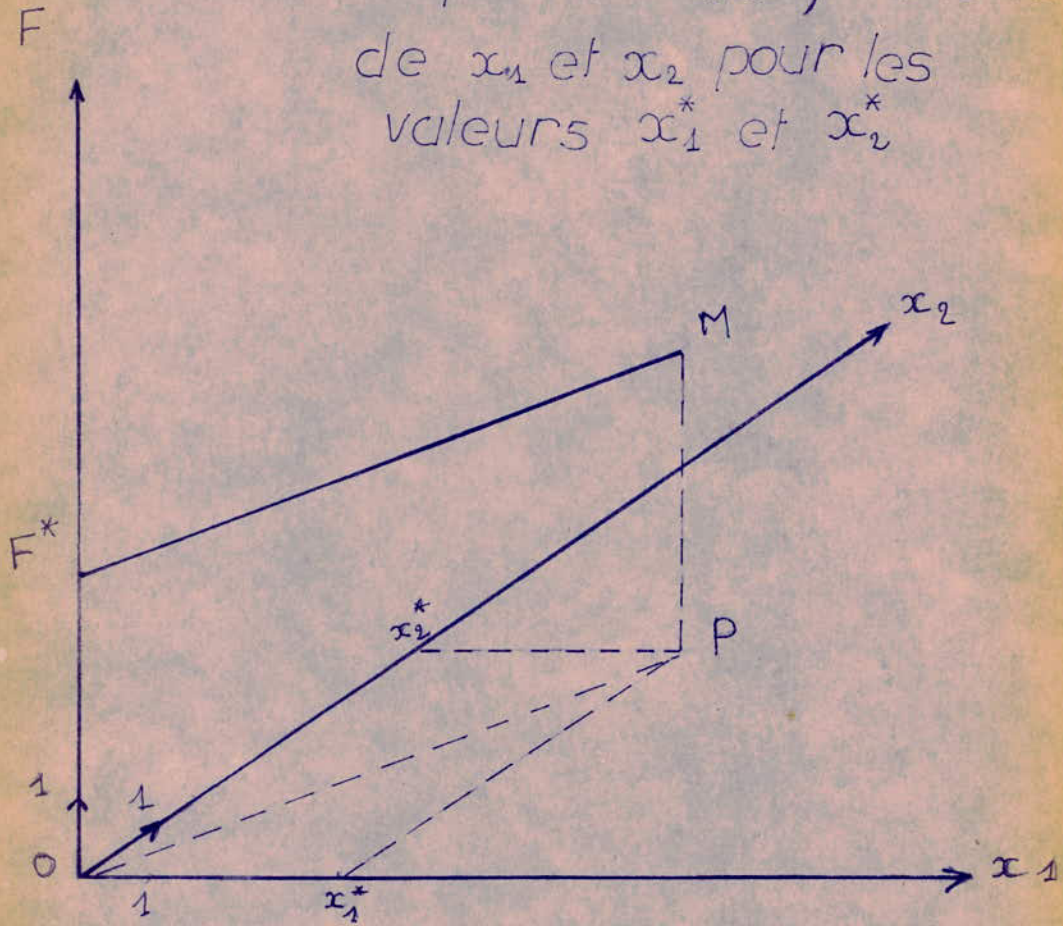
Soit une fonction de deux variables  $F(x_1, x_2)$  . A chaque couple de valeurs de  $x_1$  et  $x_2$ ,  $x_1^*$  et  $x_2^*$ , est associée une valeur  $F^*$  ( voir figure IV - a ) .

Si l'on donne à  $x_1$  et  $x_2$  des valeurs quelconques, le point P décrira le plan  $(x_1, x_2)$ , espace à deux dimensions  $E^2$  . Le point M déterminera une surface à 3 dimensions  $S$  . Il évoluera dans l'espace à 3 dimensions  $E^3$  .

C'est pourquoi, à toute fonction de deux variables, peut être généralement associée une surface à 3 dimensions .

fig IV a

Expression de  $F$ , fonction  
de  $x_1$  et  $x_2$  pour les  
valeurs  $x_1^*$  et  $x_2^*$



Supposons maintenant qu'ayant chois<sup>i</sup> des valeurs  $x_I^*$  et  $x_2^*$ , et ayant obtenu une valeur  $F^* = F(x_I^*, x_2^*)$ , l'on décide de conserver cette valeur  $F^*$  et de chercher l'ensemble des couples  $(x_I, x_2)$  tels que  $F(x_I, x_2) = F^*$ .

Cette relation entre  $x_I$  et  $x_2$  détermine une courbe plane passant par  $M$ .

C'est une ligne de niveau définie comme l'intersection de la surface à trois dimensions  $S$  et du plan passant par  $M$ , parallèle au plan  $(Ox_I, Ox_2)$ .

Plus généralement, dans le cas d'une fonction  $F = f(x_I, x_2, \dots, x_n)$  de  $n$  variables  $x_I, \dots, x_n$ , chaque suite de valeurs  $x_I^*, x_2^*, \dots, x_n^*$  détermine une valeur de la fonction  $F^* = f(x_I^*, x_2^*, \dots, x_n^*)$ .

La suite  $x_I^*, x_2^*, \dots, x_n^*$  définit comme précédemment un "point"  $P$  qui appartient cette fois à l'espace à  $n$  dimensions  $E^n$ .

A ce point  $P$  correspond un point  $M$  qui décrit une "surface" de l'espace à  $(n+1)$  dimensions  $E^{n+1}$ , on l'appelle une hypersurface.

Inversement  $f(x_I, x_2, \dots, x_n) = F^*$  est une relation entre les  $n$  variables  $x_I, x_2, \dots, x_n$  qui représente en général une surface de  $E^n$ , dite "surface de niveau".

En général, il s'agit d'optimiser une fonction  $f(x)$  de plusieurs variables  $x_I, x_2, \dots, x_n$ , appelée fonction objectif.

Les contraintes qui interviennent dans le processus définissent ce que l'on appelle le domaine admissible. Soit  $D$  ce domaine.

Les contraintes peuvent être du type :

- égalité :  $g_i(x) = 0 \quad i=1, 2, \dots, p$

- inégalité :  $h_j(x) \leq 0 \quad j=1, 2, \dots, q$

Si  $x$  appartient à  $D$ ,  $x$  est dit admissible.

Les contraintes du type égalité sont aussi appelées liaisons;



dans le cas où  $x$  est tel que  $h_j = 0$ , la contrainte inégalité d'ordre  $j$  est dite saturée.

Il serait vain de parler ici de toutes les méthodes d'optimisation. Elles diffèrent selon la nature des problèmes considérés. Nous nous intéresserons cependant uniquement aux fonctions de plusieurs variables car rares sont les systèmes ne mettant en jeu qu'une seule variable.

Les problèmes d'optimisation sont aussi appelés "problèmes de programmation".

Suivant la nature de la fonction objectif et des contraintes, on classe les problèmes de programmation en deux grandes familles :

#### IV - I - I - Programmation linéaire :

Ils sont caractérisés par une fonction objectif linéaire et des contraintes linéaires.

Un programme linéaire s'écrit, de façon générale, comme suit :

Trouver  $X$  tel que :

$L \cdot X$  soit maximum

avec  $M \cdot X \geq B$  ( ou bien  $M \cdot X \leq B$  )

ou encore : trouver  $X$  tel que :

$L \cdot X$  soit minimum

avec  $M \cdot X \geq B$  ( ou bien  $M \cdot X \leq B$  )

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad L = [\theta_1, \theta_2, \dots, \theta_n]$$

$$M = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{bmatrix}$$

$$B = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

Les  $\theta$ ,  $a$ ,  $b$  sont des nombres positifs, négatifs ou nuls.

Il s'agit, dans le premier cas par exemple, de maximiser/

$$S = \theta_1 \cdot x_1 + \theta_2 \cdot x_2 + \dots + \theta_n \cdot x_n$$

que l'on note :  $S = \sum_{i=1}^n \theta_i \cdot x_i$

sachant que :

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &\leq b_2 \\ \dots &\dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &\leq b_m \end{aligned}$$

Il est relativement aisé d'expliquer la seconde hypothèse de la maximisation et le cas de minimisation en procédant de façon analogue.

J.B. DANTZIG a proposé, il y a une trentaine d'années, la méthode du Simplexe qui permet de résoudre des problèmes de programmation linéaire.

Cette méthode consiste à progresser le long des frontières du domaine admissible, à partir d'une solution admissible prise initialement.

#### IV - I - 2 - Programmation non linéaire :

La programmation non linéaire est caractérisée par une fonction objectif linéaire ou non et des contraintes linéaires ou non, le cas où fonction objectif et contraintes sont linéaires étant exclu.

Le nombre de problèmes de programmation non linéaires est très grand. Leur résolution est relativement difficile car l'on se heurte à des difficultés mathématiques très importantes.

Nous pouvons dire que, jusqu'à présent, la théorie de la programmation non linéaire n'est pas achevée.

Un programme quelconque s'écrit :

Trouver  $x_1, x_2, \dots, x_n$  tel que :  
 $Z = f(x_1, x_2, \dots, x_n)$  soit maximum, avec  
les  $m$  inéquations ou équations :

$$\begin{aligned} \xi_i(x_1, x_2, \dots, x_n) &\leq 0 & i = 1, 2, \dots, m \\ &\text{ou } \geq 0 \\ &\text{ou } = 0 \end{aligned}$$

Il peut être aussi question de minimisation.

L'inéquation  $g_i (x_1, x_2, \dots, x_n) \leq 0$  s'écrit aussi :

$-g_i (x_1, x_2, \dots, x_n) \geq 0$  , c'est à dire :

$$g_i (x_1, x_2, \dots, x_n) \geq 0$$

On peut donc supprimer un type d'inégalité dans la formulation générale.

REMARQUES:

- 1 - la fonction objectif de n variables est quelconque .
- 2 - les contraintes à satisfaire sont quelconques.

Une nouvelle étape a été franchie il y a quelques années par H.W. KUHN et A.M. TUCKER, et, ainsi est apparue une nouvelle classe de problèmes : ce sont les problèmes de programmation quadratique.

On peut aussi envisager le cas d'une fonction quelconque à optimiser sans contraintes. Le tableau suivant résume les définitions précédentes.

Le cas de la programmation avec fonction linéaire sans contraintes est exclu, car il ne peut donner de solution finie .

fonction objectif		linéaire	quadratique	autre
		contraintes		
avec contra intes	liné- aires	programmation linéaire	programmation quadratique	programmation
	non linéai res	non linéaire avec contraintes		
sans contraintes		X		programmation non linéaire sans contraintes

REMARQUE :

Les problèmes de programmation du type général, mais pour lesquels :

- 1 - les contraintes sont uniquement des équations
- 2 - il n'y a pas d'autres restrictions sur les variables
- 3 - le nombre d'équations,  $m$ , est strictement inférieur à  $n$  : nombre de variables

4 - les fonctions  $g_i$  et  $f$  sont continues et possèdent des dérivées partielles jusqu'au second ordre au moins, sont appelés problèmes classiques d'optimisation. Ils sont solubles par d'autres méthodes que les méthodes de programmation.

Cependant, la programmation peut leur être appliquée avec profit, surtout lorsqu'on envisage de calculer sur ordinateur.

Les diverses méthodes de résolution des programmes non linéaires peuvent être classés en trois catégories :

- 1 - Les méthodes d'optimisation sans contraintes
- 2 - Les méthodes d'optimisation avec contraintes ramenant le programme à un programme non linéaire sans contraintes.
- 3 - Les autres méthodes d'optimisation , en particulier la méthode du gradient que nous développons au paragraphe suivant.

IV - 2 - LA METHODE DU GRADIENT :

C'est une méthode directe.

On appelle gradient de la fonction  $f(x)$  le vecteur ayant pour composantes  $\frac{d f(x)}{d x_i}$ .

$$d x_i$$

Le calcul du gradient peut se faire de plusieurs façons :

- algébriquement : si  $f(x)$  est connue analytiquement, l'expression algébrique de  $\frac{d f(x)}{d x_i}$  peut être déduite .

- directement si  $f(x)$  ne peut être calculé que numériquement. On utilise alors la définition de la dérivée partielle :

$$\frac{d f(x)}{d x_i} = \frac{f(x + \xi u_i) - f(x)}{\xi}$$

$u_i$  est le vecteur ayant toutes ses composantes nulles sauf la  $i$ -ème qui est égale à l'unité.

- indirectement si  $f(x)$  est lié à  $x$  par des fonctions intermédiaires.

Le principe de cette méthode est le suivant ; à partir d'un point de la surface, on se déplace dans le sens du gradient de cette fonction.

On a toujours intérêt à calculer les dérivées partielles à l'aide de leur expression analytique lorsque cela est possible.

La méthode directe du gradient ou méthode de la plus grande pente n'est pas toujours applicable.

IV - 2 - I - Méthode de la plus grande pente :

Dans le cas où les surfaces de réponses sont régulières , cette méthode permet de déterminer l'extrémum en un nombre fini d'itérations.

Supposons une surface de réponse  $Z = F ( x_1 , x_2 )$  dont les lignes de niveau sont des cercles concentriques ( figure IV.b)

En partant d'un point quelconque A, on se déplace dans la direction du gradient. Cette direction étant perpendiculaire à l'horizontale tangente en A au contour, on voit que le gradient passe par l'extrémum et la recherche ne nécessite dans ce cas qu'une seule itération ( il suffit de chercher l'extrémum dans cette direction ).

Le problème est le même dans le cas où les lignes de niveau de la surface de réponse sont des ellipses concentriques et si le point A se trouve sur l'un des axes de ces ellipses.

D'une façon générale, si une fonction d'un nombre quelconque de variables a pour contours des hypersphères concentriques, la recherche se réduit à une recherche à une seule dimension le long du gradient passant par un point quelconque, car dans ce cas, tous les gradients sont des droites qui passent par l'extrémum.

Envisageons maintenant le cas où les contours sont des ellipsoïdes concentriques ( figure IV.c ).

Si le point de départ ne se trouve pas sur l'un des axes du système d'ellipsoïdes, la recherche de l'extrémum comporte un trajet en "zig-zags", comme celui qui part de  $p_0$  et passe par  $p_2$ ,  $p_3$  et b .

Il faut remarquer qu'en principe, un calcul utilisant la méthode de la pente ne conduit à l'extrémum qu'en un nombre infini d'itérations.



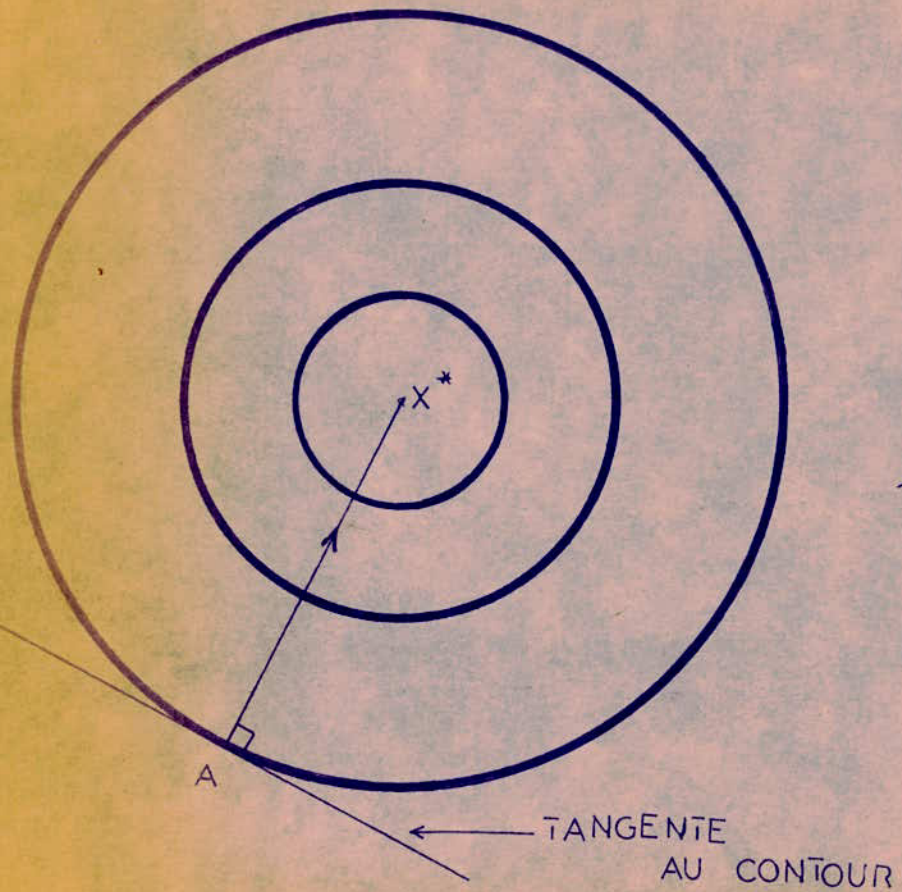


fig IV. b. CONTOURS CIRCULAIRES.

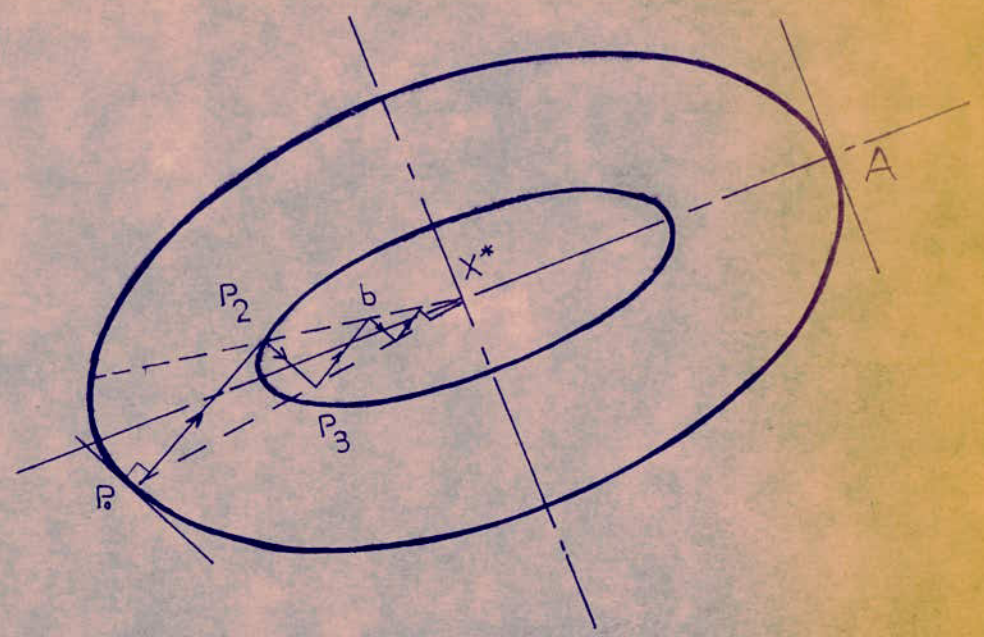


fig IV. c. CONTOURS ELLIPTIQUES.

Nous venons de voir que la méthode de la plus grande pente consiste à se déplacer dans la direction du gradient et à chercher dans cette direction l'optimum de la fonction considérée.

Dans le cas d'une fonction de  $n$  variables indépendantes, représentant une surface à  $n$  dimensions, d'équation  $F = f(x_i)$ , les relations générales qui permettent d'utiliser cette méthode sont les suivantes :

les cosinus directeurs du gradient sont :

$$l_i^j = \left\{ \frac{\partial F}{\partial x_i} / \left[ \sum_{k=1}^n \left( \frac{\partial F}{\partial x_k} \right)^2 \right]^{\frac{1}{2}} \right\}^j \quad (1)$$

les équations paramétriques de la ligne de déplacement passant par le point  $x_i^j$  seront donc, pour la détermination du point  $x_i^{j+1}$  :

$$x_i^{j+1} = x_i^j + t \cdot l_i^j \quad (2)$$

Le principe de cette méthode consiste à déterminer dans la direction définie par la relation (2) l'extrémum de la fonction. Il s'agit de déterminer  $t$  tel que :

$$F = f(x_i^{j+1}) = f(t) \text{ soit optimum} \quad (3)$$

On a donc à faire l'optimisation d'une fonction d'une seule variable. Pour cela, plusieurs techniques peuvent être utilisées.

Après avoir déterminé  $t^*$  qui rend dans la direction définie par (2) la fonction optimum, on aura la relation :

$$x_i^{j+1} = x_i^j + t^* \cdot l_i^j$$

On continue ainsi jusqu'à ce que la distance séparant deux points successifs soit compatible avec la précision demandée.

IV - 2 - 2 - Méthode de BOOTH :

C'est une variante de la méthode de la plus grande pente. On part d'un point et l'on cherche dans la direction du gradient l'optimum de la fonction.

La différence est que, au lieu de partir, pour une nouvelle itération, de cet extrémum, on part d'un point situé sur le gradient entre le point précédent et l'extrémum défini par :

$$x_i^{j+1} = x_i^j + r \cdot t \cdot l_i^j \quad \text{avec } 0 < r < 1$$

$r$  est une constante soit définie arbitrairement, soit dépendant de l'angle de deux vecteurs gradients successifs.

IV - 2 - 3 - Méthode de MARQUADT :

Cette méthode utilise des déplacements non dans la direction du gradient, mais dans une direction du gradient corrigée à chaque itération. De plus, on ne recherche plus dans ces directions corrigées l'optimum de la fonction, mais l'on s'y déplace d'un pas variable à chaque itération.

Pour une fonction de  $n$  variables indépendantes  $F = f(x_i)$ , les cosinus directeurs des directions corrigées du gradient sont données par les relations :

$$N_i^j = \left[ 1 + (x_i^j)^2 \right] \left\{ \frac{\partial F}{\partial x_i} / \left[ \sum_{k=1}^n \left[ (1 + x_k^2) \frac{\partial F}{\partial x_k} \right]^2 \right]^{\frac{1}{2}} \right\}^j$$

Dans cette direction, on se déplace d'un pas variable à chaque itération, cette valeur dépendant de l'angle des directions de deux itérations successives.

Le passage de l'itération  $j$  à l'itération  $j+1$  se fera à l'aide des relations suivantes:

$$x_i^{j+1} = x_i^j + t^j \cdot N_i^j$$

avec :  $t^j = t^{j-1} ( d_1 + d_2 \cos^4 \theta )$

$$\cos \theta = \sum_{i=1}^n N_i^j \cdot N_i^{j-1}$$

$$0 < d_1 < 1$$

$$1 - d_1 < d_2 \leq 1$$

les valeurs de  $d_1$  et  $d_2$  qui donnent satisfaction sont, d'après l'auteur :

$$d_1 = 0,5$$

$$d_2 = 1$$

REMARQUE :

Pour l'utilisation de cette méthode, il faut s'assurer qu'à chaque itération, le résultat est meilleur qu'à l'itération précédente, sinon, il faut revenir à l'itération précédente et diminuer arbitrairement la valeur de  $t^j$ .

Nous avons tenu à présenter la méthode du gradient car elle permet de calculer le réseau optimal.

En effet, nous avons défini précédemment :

$$\alpha ( m , n , k ) = \frac{d \bar{t} (m,n)}{d TI (p)} \quad \text{pour } p = P (m,n,k)$$

Le gradient de la fonction objectif est le vecteur de composantes :

$$\frac{d TTPA}{d TI (p_k)}$$

or,  $TTPA = \sum_{\substack{m=I \\ n=I}}^{N_T} OD (m,n) \cdot \bar{t} (m,n)$

$$d'où : \frac{d TTPA}{d TI (p)} = \sum_{\substack{m=I \\ n=I}}^{N_T} OD (m,n) \cdot \frac{d \bar{t} (m,n)}{d TI (p)}$$

$$\text{et } \frac{d TTPA}{d TI (p_k)} = \sum_{\substack{m=I \\ n=I}}^{N_T} OD (m,n) \cdot \frac{d \bar{t} (m,n)}{d TI (p_k)}$$

Le gradient de la fonction objectif s'écrit donc :

$$\frac{d TTPA}{d TI (p_k)} = \sum_{\substack{m=I \\ n=I}}^{N_T} OD (m,n) \cdot \alpha (m,n,k)$$

Après avoir défini le gradient de la fonction objectif, nous allons raisonner sur deux parcours et montrer ainsi comment peut être cette méthode peut être appliquée.

Considérons deux parcours  $p_I$  et  $p_2$ . Les temps d'intervalle sur ces deux parcours sont  $TI(p_I)$  et  $TI(p_2)$ , ils doivent vérifier :

$$\begin{aligned} TI(p_I) &\leq TI_{\max}(p_I) \\ TI(p_2) &\leq TI_{\max}(p_2) \end{aligned}$$

Soit  $S$  la valeur de la fonction objectif.

$$S = f(TI(p_I), TI(p_2), \dots, TI(p_T))$$

Les surfaces de réponse  $S$  sont supposées assez régulières et présentant l'aspect d'ellipsoïdes. Sur les deux parcours  $p_I$  et  $p_2$ , les surfaces de réponse sont des ellipses (fig IV.d).

Les temps de rotation sur chaque boucle sont constants. Le nombre de véhicules dont dispose la compagnie est donné, c'est une contrainte sur le processus, donc :

$$\sum_{j=I}^{B_T} \frac{T_R(j)}{TI(j)} \leq A_T$$

$TI(j)$  est le temps d'intervalle correspondant à la boucle  $j$ .

Dans l'espace à deux dimensions considéré, nous aurons :

$$\frac{T_R(b_I)}{TI(p_I)} + \frac{T_R(b_2)}{TI(p_2)} \leq K$$

$K$  étant une constante égale au nombre d'autobus circulant sur les boucles  $b_I$  et  $b_2$ , correspondant respectivement aux parcours  $p_I$  et  $p_2$ .

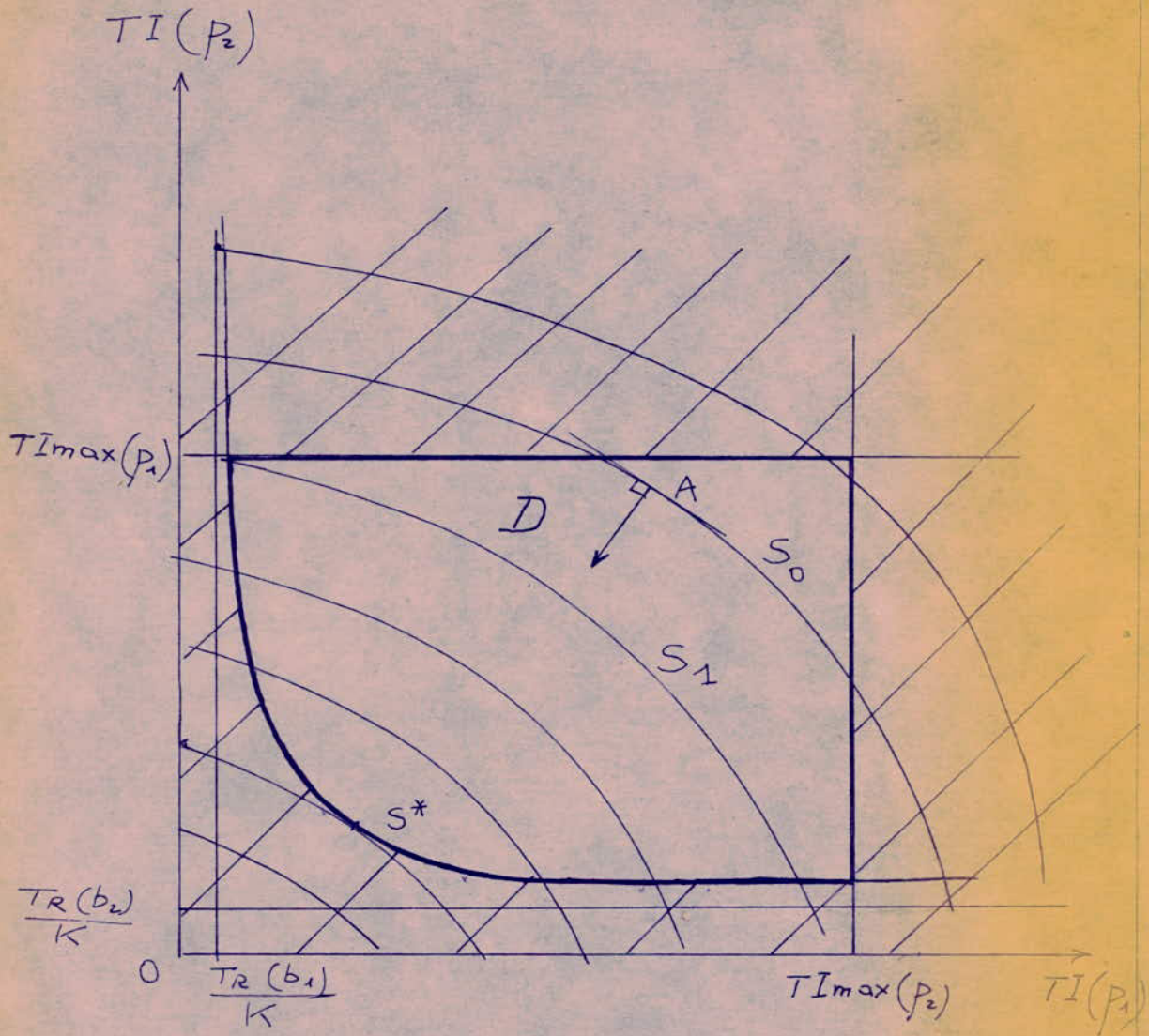


Fig IV. d. Représentation du domaine admissible D.

Cette relation exprime l'équation d'une hyperbole d'asymptotes  $\frac{T_R (b_1)}{K}$  et  $\frac{TR (b_2)}{K}$ .

Dans le cas général d'un nombre  $P_T$  de parcours, il s'agit de minimiser la valeur de la fonction objectif  $S$  en tenant compte des contraintes suivantes :

$$\sum_{b=1}^{B_T} \frac{T_R (b)}{X (b)} \leq A_T \frac{B_T}{P_T}$$

$$TI (p_i) \leq TI_{max} (p_i) \quad \text{pour } i = 1, \dots, P_T$$

Nous avons représenté sur la figure IV.d le domaine admissible  $D$ . Il s'agit d'évoluer dans ce domaine.

Dans le cas multidimensionnel, on part d'un point  $A$  où la fonction objectif a la valeur  $S_0$ , ce point doit appartenir au domaine admissible  $D$ , la direction de déplacement est donnée par le gradient de la fonction objectif  $\frac{d TPA}{d TI (p_k)}$ ,

cependant, la difficulté réside dans le choix du pas de déplacement. Ce choix est fonction de la précision voulue. Pour une précision moyenne, nous pouvons prendre pour pas de cheminement sur chaque axe  $TI (p_i)$  la valeur  $\frac{TI (p_i)}{100}$ . On arrive ainsi

à un nouveau point  $B$ , en ce point, on calcule la nouvelle valeur du gradient, la valeur de la fonction objectif en ce point est  $S_I$ . On continue ainsi jusqu'à l'optimum  $S^*$ .



V - CONCLUSION :

Nous estimons avoir atteint notre objectif au cours de cette étude, à savoir, nous avons établi un certain nombre d'algorithmes permettant la description du processus durant une certaine tranche horaire, nous avons présenté toutes les étapes qui permettent de calculer la fonction objectif et finalement nous donnons une application de la méthode du gradient au problème considéré.

Cependant, cette étude ne prétend pas être complète et applicable directement à la simulation d'un réseau de transport urbain réel. En effet, nous avons été amené à simplifier les caractéristiques et à ne considérer que ce qui était important pour la simulation. La simulation d'un réseau réel nécessite la prise en compte des données relatives à une journée entière, depuis le départ du premier autobus le matin jusqu'à la rentrée du dernier le soir. De plus, les données ne sont pas les mêmes pour tous les jours de la semaine.

La journée doit être découpée en un certain nombre  $T$  de tranches horaires correspondant par exemple :

- pour  $t = 1$  à la tranche creuse du matin,
- pour  $t = 2$  à l'heure de pointe du matin,
- pour  $t = 3$  à la tranche creuse d'avant midi,
- .....

$t$  désignant l'indice des tranches.

Les grandeurs aléatoires seront supposées suivre des lois dont les paramètres statistiques sont constants à l'intérieur de ces tranches. Ces paramètres ont donc, en fonction du temps,

une allure de créneaux.

Cependant, il faut permettre un passage souple d'une tranche à la tranche suivante.

L'arrivée des usagers à une station a été supposée régulière, en réalité, elle dépend des fréquences de passage des autobus sur les différentes lignes du réseau. Si la fréquence sur une ligne est suffisamment élevée, la notion d'horaire n'influe pas sur le comportement des usagers. Sinon, l'arrivée des usagers, qui est un paramètre aléatoire difficile à évaluer, dépend du réseau à simuler. Si les heures de passage des véhicules sont connus par les usagers, ceux-ci ne se présentent à la station que quelques instants seulement avant le départ des autobus, il y a donc là réduction du temps moyen d'attente et par conséquent, réduction du temps de trajet moyen.

Les tests effectués pour déterminer l'indicateur de possibilités sont donnés pour un cas d'exploitation normale. Les hypothèses sont donc remises en cause en cas de perturbation (lorsqu'il pleut par exemple).

La détermination de la matrice demande origine-destination pour chaque parcours doit se faire à partir d'enquêtes statistiques; il en est de même pour la matrice des temps de marche à pied. Cette dernière peut également être déduite à partir des distances entre les différents noeuds  $D(m,n)$  et d'une vitesse moyenne  $V$  de marche à pied.

Il est évident que la méthode proposée ne permet pas directement la simulation d'un réseau de transport important comportant un grand nombre de noeuds car les matrices seraient trop encombrantes. Il est nécessaire dans ces cas de diviser le réseau en un certain nombre de secteurs et d'étudier les divers secteurs.

## BIBLIOGRAPHIE

---

1 - Hervé SEGALEN. Michel JOUVENT

La programmation non linéaire

( Dunod Economie 1971 )

2 - Jean VIGNES

Thèse de doctorat : Etude et mise en oeuvre d'algorithmes  
de recherche d'un extrémum d'une fonction  
de plusieurs variables.

( Faculté des sciences de l'université de  
Paris . 1969 )

3 - V. KAFAROV

Méthodes cybernétiques et technologie chimique

( Editions de MOSCOU )

4 - Annexe Simulation du projet :

Etude des possibilités d'aide électronique aux trans-  
ports en commun de la ville de Toulouse.

( Département Automatique du Centre d' Etudes  
et de Recherches de l'Ecole Nationale Supérieure  
de l'Aéronautique et de l'Espace de Toulouse.  
juillet 73 ).

5 - D.J. WILDE

Méthodes de recherche d'un optimum ( Dunod 1966 )

6 - R. BOUDAREL - J. DELMAS - P. GUICHET .

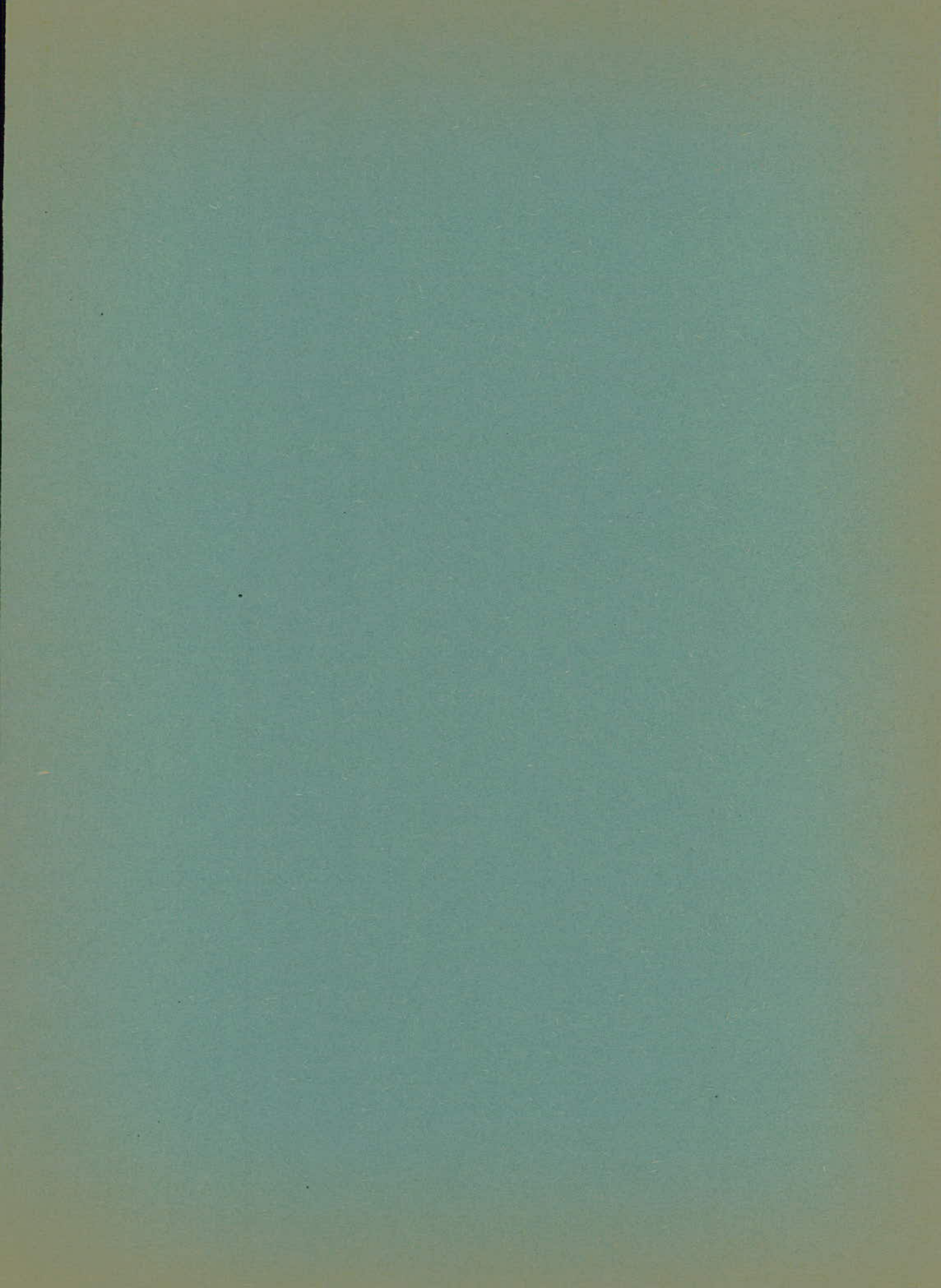
Commande optimale des processus ( tome II )  
( Dunod 68 )

7 - Ph. BENDTSEN et T. ROLLIS

" Transport en dehors des villes"

<sup>è</sup>  
4<sup>e</sup> symposium international sur la théorie et la prati-  
que dans l'économie des transports.

LA HAYE. 5-7 Oct. 1971



- A N N E X E -

-----

	page
I. DONNEES NUMERIQUES	I
II. LE PROGRAMME DE SIMULATION	5
III. ANALYSE ET INTERPRETATION DES RESULTATS	18
IV. CONCLUSION	36

SIMULATION DU RESEAU IMAGE  
-----

Nous avons considéré, pour valider le modèle de simulation que nous proposons, un réseau urbain d'autobus, relativement simple, purement imaginaire, que nous avons appelé arbitrairement réseau IMAGE.

Ce réseau est représenté sur la figure I. Nous avons essayé, dans le choix des données numériques, d'être aussi proches que possible de la réalité.

La simulation de ce réseau a été réalisée sur l'ordinateur CII I0070 du Centre d'Etudes et de Recherche en Informatique de Oued-Smar.

I - DONNEES NUMERIQUES :

Le réseau IMAGE comporte :

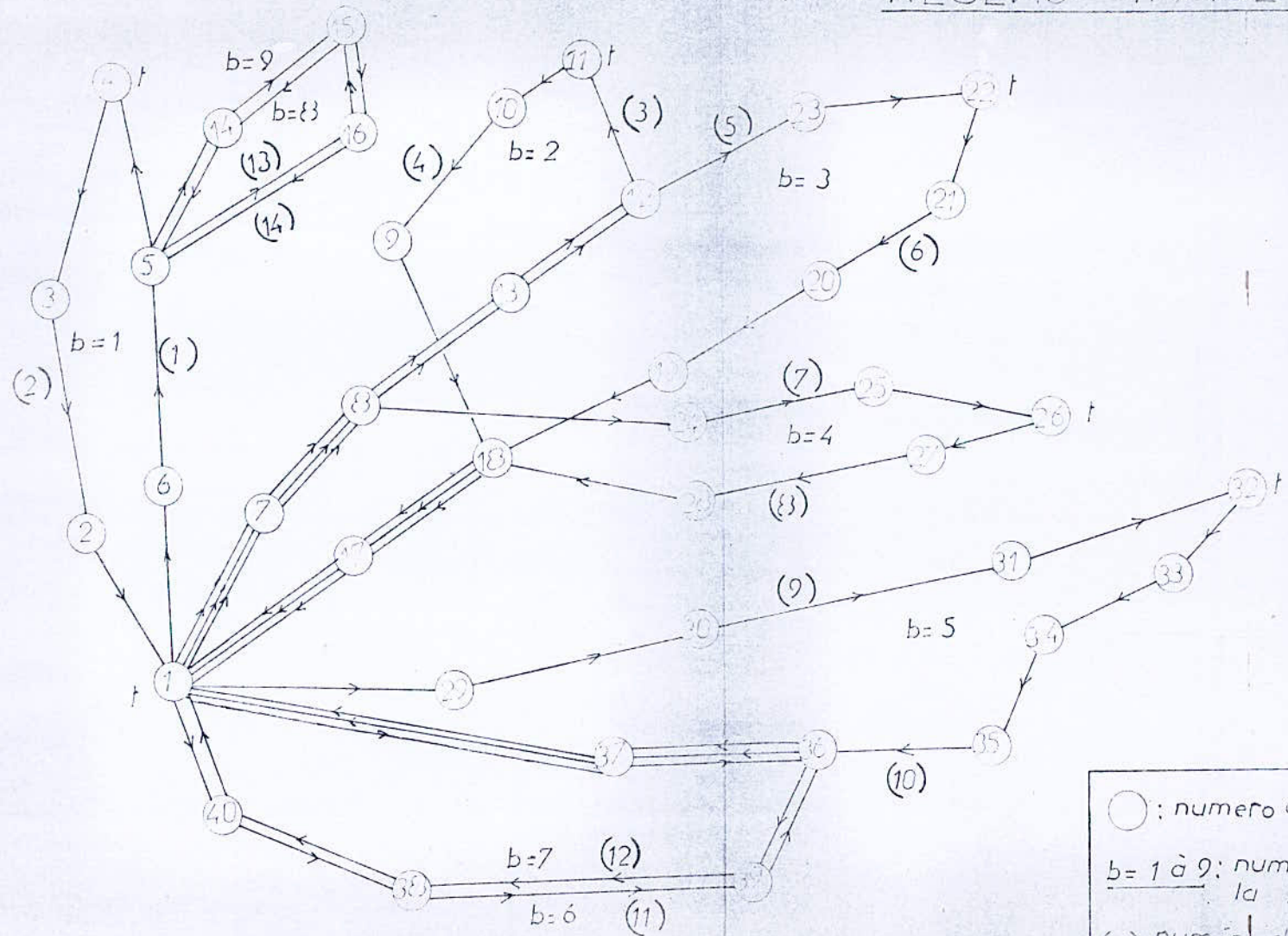
- 40 noeuds, numérotés de I à 40
- 7 lignes : 5 lignes aller-retour et 2 lignes circulaires
- 9 boucles

Les sens de parcours sont indiqués par des flèches. Les noeuds n° 4 , I , II , 22 , 26 et 32 sont des terminus, le noeud n° 5 est un terminus particulier, il ne l'est que pour les parcours I3 et I4. Les terminus sont indiqués par la lettre t sur la figure.

Nous avons simulé le réseau IMAGE sur une tranche horaire de deux heures, comprise entre 9 et II heures.

L'unité de temps est la minute.

# Fig. 1. RESEAU IMAGE



○ : numero du nœud  
b = 1 à 9 : numero de la boucle  
 ( ) : numero du Parcours

Nous avons pris, vu les caractéristiques du réseau

TREPOS = 10 mn  
NBMSTA = 7  
NLMAX = 3  
NBMNA = 5  
NBTBUS = 50  
NBMPPB = 100

La matrice de demande OD (m,n) est donnée pour m et n variant de 1 à 40, sur le tableau 2.

La matrice des temps de marche à pied TMP (m,n) est représentée sur le tableau 3.

A partir de la figure I, nous pouvons déduire la matrice ligne NBSTA (L) , pour L variant de 1 à 14.

L	I	2	3	4	5	6	7	8	9	10	11	12	13	14
NBSTA(L)	4	4	6	6	7	7	6	6	5	7	7	7	5	5

Tableau n° 4.

De même, nous pouvons déduire les matrices  $N_A (M,K)$  et NEULI ( I , L ).



(TABLEAU 2) IOD (M.N)

(tranche horaire comprise entre 0h et 1h)

2bis

M \ N	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	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1	0	90	85	100	110	120	82	87	65	75	35	130	10	105	100	82	68	76	35	42	40	17	23	56	68	76	40	18	32	44	31	22	17	25	16	40	10	29	20	27
2	87	0	45	95	107	99	84	95	58	73	41	102	17	85	86	77	108	87	56	45	33	14	27	66	74	55	37	25	59	6	35	24	15	19	31	19	14	46	29	22
3	56	89	0	100	53	47	87	76	87	56	77	99	67	109	110	77	76	52	77	42	19	27	32	76	54	87	26	21	66	6	34	27	19	12	22	27	7	49	25	31
4	89	78	43	0	27	19	23	33	46	44	82	29	99	77	88	67	70	63	39	28	12	37	25	81	62	44	52	88	51	74	41	23	27	34	41	55	6	47	33	44
5	47	61	26	71	0	26	29	45	32	49	56	33	67	21	44	79	35	70	40	23	34	120	116	19	67	37	66	70	47	4	89	17	47	19	57	32	7	29	17	57
6	79	78	49	39	104	0	57	71	23	63	55	47	55	96	67	81	131	87	47	38	27	42	31	24	87	58	57	71	53	12	97	36	43	34	77	19	73	35	21	70
7	22	36	33	11	27	39	0	73	53	65	34	46	60	21	38	149	45	37	52	36	31	39	53	19	95	57	42	66	56	2	89	47	39	35	81	24	73	33	19	47
8	27	35	42	17	19	23	30	0	41	37	51	33	42	26	39	23	47	33	71	50	37	32	19	79	87	57	40	67	55	3	90	27	50	19	71	26	15	41	33	55
9	56	37	44	18	27	37	41	81	0	39	47	43	58	33	41	56	77	67	45	39	26	27	21	55	66	37	31	53	37	5	51	37	67	36	56	19	19	22	31	13
10	29	41	53	36	51	42	39	88	52	0	31	53	92	45	27	24	29	31	43	51	37	39	23	58	77	35	42	71	39	55	47	42	50	32	53	22	21	27	23	21
11	30	19	31	22	27	19	27	35	47	59	0	42	46	33	29	37	29	19	21	18	32	41	32	23	25	31	30	40	37	21	25	27	23	32	29	18	7	31	29	37
12	37	23	42	31	29	31	35	31	36	44	56	0	44	39	54	29	41	27	31	30	27	43	25	30	28	33	31	47	32	29	28	29	31	37	4	27	59	44	36	23
13	21	20	44	37	27	34	38	27	51	59	63	44	0	57	48	44	36	33	41	37	26	41	28	30	20	38	35	45	30	31	25	28	30	41	27	39	44	29	25	
14	23	110	53	39	25	36	116	19	53	57	65	41	49	0	53	47	43	37	44	33	27	45	39	44	27	23	37	40	44	32	29	33	47	27	25	20	19	27	23	32
15	31	17	22	31	27	24	112	17	34	37	51	49	43	29	0	31	44	39	41	37	32	51	32	41	29	50	35	37	47	25	27	40	29	27	29	12	37	31	27	
16	27	19	37	22	29	31	117	121	39	27	49	53	77	53	77	0	59	71	77	87	82	76	59	97	53	71	73	57	44	0	57	39	64	56	37	53	-2	87	34	29
17	31	42	53	51	47	67	39	48	51	54	63	91	87	57	82	64	0	69	93	97	89	77	63	81	67	73	66	81	88	0	59	47	79	81	59	62	10	99	63	57
18	63	41	86	47	69	72	53	77	66	51	74	70	89	53	81	73	63	0	88	86	91	84	71	82	65	76	79	91	87	38	69	57	89	80	64	62	11	93	66	39
19	41	59	85	81	67	71	59	86	77	62	85	69	72	77	63	88	72	78	0	71	88	71	82	73	74	85	91	77	83	17	72	75	69	87	71	69	13	79	71	37
20	31	67	76	72	76	80	68	77	81	70	76	59	77	62	74	53	87	79	73	0	79	62	73	64	83	76	79	55	61	33	60	72	63	91	60	62	13	76	73	41
21	39	76	85	61	77	61	53	59	73	69	75	59	83	99	41	47	53	81	88	79	0	69	82	73	74	58	73	66	72	65	71	81	57	57	57	57	69	46	49	
22	48	73	53	62	87	72	47	52	57	58	69	67	72	81	77	64	71	92	46	67	0	70	72	59	63	87	91	87	93	79	89	89	79	89	89	87	82	57	54	63
23	35	59	71	82	91	83	67	56	64	67	83	72	80	87	91	87	80	88	77	73	71	0	87	73	97	87	91	87	69	71	86	99	83	92	73	87	93	73	61	66
24	71	61	87	73	79	71	65	83	67	81	90	77	83	89	87	97	72	87	83	97	87	93	73	0	87	91	79	71	85	91	79	89	97	87	83	70	82	79	87	89
25	53	77	39	91	78	69	83	65	78	80	87	83	82	77	74	73	79	91	79	88	89	99	82	93	0	79	90	83	87	94	73	69	77	83	69	72	91	75	87	63
26	61	73	67	79	81	78	73	93	92	97	89	79	87	91	73	65	91	69	83	87	89	91	95	79	87	0	83	88	99	97	93	71	83	85	90	77	88	77	97	76
27	77	89	76	78	87	79	87	95	92	93	79	80	59	76	93	79	95	71	73	88	89	95	77	83	86	88	0	69	75	95	84	87	79	77	82	87	92	79	81	87
28	53	59	79	91	93	88	79	83	87	99	92	76	65	88	85	91	96	87	89	85	93	87	79	90	91	93	87	0	77	91	73	97	98	79	87	91	94	76	65	79
29	77	81	79	77	99	97	92	87	81	85	73	87	93	84	79	59	93	79	87	86	77	99	93	95	77	86	89	90	0	79	77	84	91	87	93	95	79	78	39	96
30	61	97	89	82	73	79	87	89	99	100	131	94	105	96	97	111	121	87	63	79	98	96	107	103	91	89	119	79	93	0	89	102	99	87	98	105	100	89	69	79
31	73	102	97	97	89	109	117	124	97	88	99	117	123	103	109	87	89	83	115	29	99	97	101	102	97	79	86	81	101	77	0	87	103	97	99	78	87	93	114	89
32	77	90	103	102	83	79	87	97	108	87	93	99	104	117	97	79	87	89	97	99	96	108	97	78	93	96	103	81	102	94	27	0	91	97	87	67	71	73	79	81
33	65	77	89	91	78	69	95	93	100	93	97	88	87	79	61	87	91	99	103	79	84	86	98	69	84	106	92	92	90	87	107	102	0	59	63	77	13	77	67	87
34	76	97	98	79	87	91	93	67	91	84	88	99	73	87	79	89	101	109	97	87	89	106	96	79	74	94	102	72	80	76	87	82	81	0	79	96	101	103	91	73
35	76	95	96	80	88	92	35	77	82	85	98	99	83	107	97	90	100	97	87	90	91	96	76	89	104	97	92	82	87	87	107	102	97	89	0	99	13	79	89	87
36	67	96	97	81	89	93	96	78	93	86	95	90	94	97	107	97	101	102	89	93	84	92	79	93	101	97	93	84	97	86	89	104	87	109	101	0	79	89	109	78
37	83	94	93	101	109	86	86	88	83	79	83	80	74	87	117	81	98	92	93	83	84	93	82	91	99	79	96	86	107	8	89	94	87	101	103	89	0	79	99	88
38	72	95	94	83	79	93	106	88	93	89	94	92	85	91	87	85	94	87	99	89	100	97	79	92	98	79	87	89	97	8	89	97	107	91	93	87	19	0	103	79
39	63	87	85	99	101	104	76	88	91	87	93	87	87	93	100	117	79	81	93	87	92	103	97	93	87	82	83	88	97	115	79	87	98	108	109	96	17	79	0	73
40	57	97	98	89	73	104	113	131	127	98	97	113	118	141	117	79	87	79	89	99	101	106	89	98	95	106	97	127	101	79	99	93	104	79	89	93	18	109	89	0

La matrice NEULI (I,L) est donnée par le tableau 7.

NEULI ( I , L )

L I	I	2	3	4	5	6	7	8	9	10	11	12	13	14
1	I	4	I	II	I	22	I	26	I	32	I	I	5	5
2	6	3	7	10	7	21	7	27	29	33	40	37	16	14
3	5	2	8	9	8	20	8	28	30	34	38	36	15	15
4	4	I	I3	I8	I3	I9	24	I8	31	35	39	39	I4	I6
5	0	0	I2	I7	I2	I8	25	I7	32	36	36	38	5	5
6	0	0	II	I	23	I7	26	I	0	37	37	40	0	0
7	0	0	0	0	22	I	0	0	0	I	I	I	0	0

Tableau 7.

Les matrices NONA (M,K) et TPBIN (M,K) sont donnés sur les tableaux 5 et 6.

NONA ( M , K )

M <sup>K</sup>	I	2	3	4	5
I	6	7	29	37	40
2	I	0	0	0	0
3	2	0	0	0	0
4	3	0	0	0	0
5	4	I4	I6	0	0
6	5	0	0	0	0
7	8	0	0	0	0
8	I3	24	0	0	0
9	I8	0	0	0	0
I0	9	0	0	0	0
II	I0	0	0	0	0
I2	II	23	0	0	0
I3	I2	0	0	0	0
I4	5	I5	0	0	0
I5	I4	I6	0	0	0
I6	5	I5	0	0	0
I7	I	0	0	0	0
I8	I7	0	0	0	0
I9	I8	0	0	0	0
20	I9	0	0	0	0
2I	20	0	0	0	0
22	2I	0	0	0	0
23	22	0	0	0	0
24	25	0	0	0	0
25	26	0	0	0	0
26	27	0	0	0	0
27	28	0	0	0	0
28	I8	0	0	0	0
29	30	0	0	0	0
30	3I	0	0	0	0
3I	32	0	0	0	0
32	33	0	0	0	0
33	34	0	0	0	0
34	35	0	0	0	0
35	36	0	0	0	0
36	37	39	0	0	0
37	36	I	0	0	0
38	39	40	0	0	0
39	36	38	0	0	0
40	I	38	0	0	0

TABLEAU 5.

TPBIN ( M , K )

I	2	3	4	5
5	4	9	8	6
4	0	0	0	0
9	0	0	0	0
I0	0	0	0	0
I2	I0	20	0	0
I0	0	0	0	0
5	0	0	0	0
7	I5	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
6	5	0	0	0
7	0	0	0	0
3	5	0	0	0
5	5	0	0	0
20	5	0	0	0
8	0	0	0	0
6	0	0	0	0
5	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
5	0	0	0	0
I3	0	0	0	0
I6	0	0	0	0
I8	0	0	0	0
I0	0	0	0	0
8	0	0	0	0
7	0	0	0	0
8	0	0	0	0
I0	0	0	0	0
4	0	0	0	0
5	0	0	0	0
5	0	0	0	0
4	0	0	0	0
7	5	0	0	0
7	8	0	0	0
I3	8	0	0	0
5	I3	0	0	0
6	8	0	0	0

TABLEAU 6.

II - LE PROGRAMME DE SIMULATION:

```

1.      C PROGRAMME PRINCIPAL
2.      COMMON/DESC/NEULI(7,I4),NOSTA(40,I4)
3.      COMMON/COMP/ICAS(40,40),LMN(40,40,3),NLMN(40,40)
4.      COMMON/CONS/NBTNEU,NBTPAR,NBOUC
5.      DIMENSION NONA(40,5),TPBIN(40,5),TPBUS(9,9,40),
6.      2TPTT(9),TI(I4),X(9),TMP(40,40),TMN(40,40),PEUS(40,40),
7.      3PROB(40,40,3),IOD(40,40),TIMAX(9)
8.      INTEGER TPBIN,TPBUS,TPTT,TMP,TREPOS,TRANCH
9.      C LECTURE ET ECRITURE DES DONNEES
10.     READ(IO5,IO) TREPOS,TRANCH
11.     READ(IO5,I20) NBTNEU,NBTPAR,NBMSTA,NLMAX,NBMNA,NBTBUS,
      NBMPPB,ITEST
12.     READ(IO5,IO) ((IOD(M,N),N=I,NBTNEU),M=I,NBTNEU)
13.     READ(IO5,98) (NBSTA(L),L=I,NBTPAR)
14.     READ(IO5,IO) ((NONA(M,K),K=I,NBMNA),M=I,NBTNEU)
15.     READ(IO5,IO) ((TPBIN(M,K),K=I,NBMNA),M=I,NBTNEU)
16.     READ(IO5,95) ((NEULI(I,L),L=I,NBTPAR),I=I,NBMSTA)
17.     READ(IO5,IO4,END=5) ((TMP(M,N),N=I,NBTNEU),M=I,NBTNEU)
18.     DO 2 N=I,NBTNEU
19.     DO 2 M=I,NBTNEU
20.     2 TMP(M,N)=TMP(N,M)
21.     WRITE(IO8,20)
22.     WRITE(IO8,90) ((IOD(M,N),N=I,NBTNEU),M=I,NBTNEU)
23.     WRITE(IO8,203)
24.     WRITE(IO8,IO) ((NONA(M,K),K=I,NBMNA),M=I,NBTNEU)
25.     WRITE(IO8,202)
26.     WRITE(IO8,IO) ((TPBIN(M,K),K=I,NBMNA),M=I,NBTNEU)
27.     WRITE(IO8,204)
28.     WRITE(IO8,98) (NBSTA(L),L=I,NBTPAR)
29.     WRITE(IO8,205)
30.     WRITE(IO8,95) ((NEULI(I,L),L=I,NBTPAR),I=I,NBMSTA)
31.     WRITE(IO8,206)
32.     WRITE(IO8,IO6) ((TMP(M,N),N=I,NBTNEU),M=I,NBTNEU)
33.     CALL PARAM(NBMNA,TREPOS,NONA,TPBIN,TPBUS,TPTT)
34.     DO 3 NL=I,NBOUC
35.     NBUT=IFIX(0.8*FLOAT(NBTBUS))+I
36.     NMBUS=IFIX(FLOAT(NBUT)/FLOAT(NBOUC))+I
37.     X(NL)=FLOAT(TPTT(NL))/FLOAT(NMBUS)
38.     3 CONTINUE
39.     CALL TIMI(K,TI)
40.     CALL TRANCH(NLMAX,TMP,TM,TI,PEUS,PROB,TPBUS,IW)
41.     CALL AFFECT(NBMSTA,NBMPPB,TM,IOD,PEUS,PROB,TRANCH,
42.     2TIMAX,TPTT,NBTBUS,IW)
43.     90 FORMAT(40 I3)
44.     95 FORMAT(I4 I4)
45.     98 FORMAT(I4 I4)
46.     IO0 FORMAT(20 I4)
47.     IOI FORMAT(5 I4)
48.     IO4 FORMAT(40 I2)
49.     IO6 FORMAT(40 I3)
50.     IIO FORMAT(IK,I5,IX,I5)

```

```
51.      I20 FORMAT(8 I4)
52.      201 FORMAT(20X,I7H TABLEAU IOD(M,N))
53.      202 FORMAT(20X,I9H TABLEAU EPBIN(M,K))
54.      203 FORMAT(20X,I8H TABLEAU NONA(M,K))
55.      204 FORMAT(29H NOMBRE DE STATIONS PAR LIGNE)
56.      205 FORMAT(20X,I9H TABLEAU EBULI(I,L))
57.      206 FORMAT(20X,I7H TABLEAU TTE(M,N))
58.      IF(ITEST.EQ.0)STOP 9
59.      STOP
60.      END
```

```
I.      SUBROUTINE PARAM(NEMNA, TREPØS, NØNA, TPBIN, TPBUS, TPPTT)
2.      CØMMØN/DESC/NEULI(7, I4), NØSTA(40, I4)
3.      CØMMØN/CØNS/NBTNEU, NBTPAR, NBØUC
4.      DIMENSION NØNA(40, 5), TPBIN(40, 5), TPBUS(9, 9, I4), TPPTT(9)
5.      INTEGER TPBIN, TPBUS, TPPTT, TREPØS
6.      C INITIALISATIØN
7.      NL=I
8.      TPPTT(NL)=I
9.      NDEP=NEULI(I, I)
10.     DØ I L=I, NBTPAR
11.     DØ I M=I, NBTNEU
12.     I NØSTA(M, L)=0
13.     DØ 50 L=I, NBTPAR
14.     N=NEULI(I, L)
15.     NS=NBSTA(L)
16.     NSI=NBSTA(L)-I
17.     DØ 3 I=I, NSI
18.     M=N
19.     II=I+I
20.     N=NEULI(II, L)
21.     C CALCUL DE NØSTA(M, L)
22.     NØSTA(M, L)=I
23.     DØ 30 KN=I, NEMNA
24.     IF(NØNA(M, KN).EQ.N) GØ TØ 4
25.     30 CØNTINUE
26.     STØPI
27.     C CALCUL DU TEMPS DE PARCØURS ENTRE 2 STATIØNS CØNSECUTIVES
28.     4 TPBUS(I, II, L)=TPBIN(M, KN)
29.     3 CØNTINUE
30.     IF(N.NE.NEULI(I, L)) NØSTA(N, L)=NS
31.     NS2=NS-2
32.     DØ 7 I=I, NS2
33.     J2=I+2
34.     DØ 7 J=J2, NS
```

```
35.      TPBUS(I,J,L)=0
36.      NTR=J-I
37.      C CALCUL DU TEMPS DE PARCOURS ENTRE 2 STATIONS QUELCONQUES
38.      DO 7 K=I,NTR
39.      7 TPBUS(I,J,L)=TPBUS(I,J,L)+TPBUS(I+K-I,I+K,L)
40.      C CALCUL DE TPTT(NL)
41.      TPTT(NL)=TPTT(NL)+TPBUS(I,NS,L)+TREPOS
42.      IF(N.NE.NEULI(I,L)) GO TO 9
43.      DO 10 I=2,NSI
44.      10 TPBUS(I,I,L)=TPBUS(I,NS,L)
45.      9 CONTINUE
46.      IF(L.EQ.NBTPAR) GO TO I5
47.      IF(NDEP.NE.N) GO TO 50
48.      NL=NL+I
49.      TPTT(NL)=0
50.      C TEST SUR PARCOURS: ALLER ? RETOUR ? CIRCULAIRE ?
51.      NDEP=NEULI(I,L+I)
52.      50 CONTINUE
53.      I5 IF(NDEP.NE.N) STOP2
54.      NBOUC=NL
55.      WRITE(108,207)
56.      WRITE(108,60) ((NOSTA(M,L),L=I,NBTPAR),M=I,NBTREPOS)
57.      WRITE(108,208)
58.      WRITE(108,70) (TPTT(NL),NL=I,NBOUC)
59.      60 FORMAT(I4I6)
60.      70 FOTMAT(IX,9I5)
61.      207 FORMAT(20X,'TABLEAU NOSTA(I,L)')
62.      208 FORMAT(10X,'TEMPS DE ROTATION TPTT(NL)')
63.      RETURN
64.      END
```

```
1. C TEMPS D'INTERVALLES INITIAUX
2.   SUBROUTINE TIINI(X, TI)
3.   COMMON/DESC/NEULI(7, I4), NBSTA(I4), NOSTA(40, I4)
4.   COMMON/CONS/NBTNEU, NBTPAR, NBOUC
5.   DIMENSION X(9), TI(I4)
6. C INITIALISATION
7.   NL=I
8.   NDEP=NEULI(I, I)
9.   DO I L=I, NBTPAR
10.  TI(L)=X(NL)
11.  IF(NDEP.NE.NEULI(NBSTA(L), L)) GO TO I
12.  NL=NL+I
13.  IF(L.EQ.NBTPAR) GO TO I
14.  NDEP=NEULI(I, L+I)
15. I CONTINUE
16. RETURN
17. END
```



```
1. C ETUDE DES TRAJETS DIRECTS
2. SUBROUTINE TRAMOY(NLMAX,TMP,TTM,TI,PBUS,PROB,TPBUS,IW)
3. COMMON/DESC/NEULI(7,I4),NBSTA(I4),NOSTA(40,I4)
4. COMMON/COMP/ICAS(40,40),LMN(40,40,3),NLMN(40,40)
5. COMMON/CONS/NBTNEU,NBTPAR,NBOUC
6. DIMENSION TMP(40,40),TTM(40,40),TI(I4),PBUS(40,40),
   PROB(40,40,3)
7. DIMENSION TPBUS(I2,I2,I4),TISL(I4)
8. INTEGER,R,TPBUS TMP
9. DO 1M=1,NBTNEU
10. DO 2N=1,NBTNEU
11. IF(M.NE.N) GO TO 3
12. TTM(M,N)=0.
13. GO TO 2
14. 3 CALL LIGNES(MMN,NLMAX,TPBUS,TMP,TI,TIM)
15. IF(ICAS(M,N).LE.2) GO TO 6
16. C TOUT LE MONDE PART A PIED
17. TTM(M,M)=FLOAT(TMP(M,N))
18. GO TO 2
19. 6 NLC=NLMN(M,N)
20. NLCL=NLC-1
21. Y=0.
22. DO 7 IP=1,NLC
23. L=LMN(M,N,IP)
24. PPL=1.
25. PPLL=0.
26. IF(NLC.EQ.1) GO TO 9
27. IL=0
28. DO 8R=1,NLC
29. IF(R.EQ.IP) GO TO 8
30. IL=IL+1
31. TISL(IL),TI(LMN(M,N,R))
32. 8 CONTINUE
33. DO 10R=1,NLCL
34. PPLL=PPLL+(COFPOL(R,NLCL,TISL)*(-TIM)**R)/(1+R)
35. 10 CONTINUE
36. PPL=PPLL
37. 8 CONTINUE
38. C CALCUL DE PROB(M,N,IP)
39. PROB(M,N,IP)=PPL
40. Y=Y+PPL*FLOAT(TPBUS(NOSTA(M,L),NOSTA(N,L),L))
41. 7 CONTINUE
42. TISL(NLC)=TI(L)
43. V=-TIM
44. IF(ICAS(M,N).NE.2) GO TO 11
45. V=Y-FLOAT(TMP(M,N))
46. 11 FR=0.
47. PR=0.
```

```
48.      DO I2IP=1,NLC
49.      CN=COFPOL(IP,NLC,TISL)
50.      CK CN*V**IP
51.      FR=FR-CK
52.      PR=PR-CK*FLOAT(IP)/FLOAT(IP+1)
53.      I2 CONTINUE
54.      C CALCUL DE PBUS(M,N)
55.      PBUS(M,N)=FR
56.      ATA=-V*FR
57.      IF(ICAS(M,N).EQ.2)Y=FLOAT(TMP(M,N))+V*FR
58.      TTM(M,N)ATA+Y
59.      2 CONTINUE
60.      1 CONTINUE
61.      IW=1
62.      IF(IW.NE.1)RETURN
63.      WRITE(I08,209)
64.      WRITE(I08,I07)((ICAS(M,N),N=1,NBTNEU),M=1,NBTNEU)
65.      WRITE(I08,2I0)
66.      WRITE(I08,II3)((NLMN(M,N),N=1,NBTNEU),M=1,NBTNEU)
67.      WRITE(I08,2II)
68.      WRITE(I08,I09)((LMN(M,N,IP),IP=1,NLMAX),N=1,NBTNEU),
M=1,NBTNEU)
69.      WRITE(I08,2I2)
70.      WRITE(I08,II0)((PROB(M,N,IP),IP=1,NLMAX),N=1,NBTNEU),
M=1,NBTNEU)
71.      WRITE(I08,2I3)
72.      WRITE(I08,III)((PBUS(M,N),N=1,NBTNEU),M=1,NBTNEU)
73.      I07 FORMAT(40 I3)
74.      II3 FORMAT(40 I3)
75.      I09 FORMAT(40 I3)
76.      II0 FORMAT(20F5.3)
77.      III FORMAT(20F5.3)
78.      209 FORMAT('INDICATEUR DE POSSIBILITES ICAS(M,N)')
79.      2I0 FORMAT('NOMBRE DE PARCOURS CONCURRENTS NLMN(M,N)')
80.      2II FORMAT('NUMEROS DES PARCOURS CONCURRENTS')
81.      2I2 FORMAT('REPARTITION DES USAGERS SUR LES PARCOURS
CONCURRENTS')
82.      2I3 FORMAT(10X,18H TABLEAU PBUS(M,N))
83.      RETURN
84.      END
```

```
1. C RECHERCHE DES LIGNES CONCURRENTES
2.   SUBROUTINE LIGNES(M,N,NLMAX,TPBUS,TMP,TI,TIM)
3.   COMMON/DESC/NEULI(7,I4),NBSTA(14),NOSTA(40,14)
4.   COMMON/COMP/ICAS(40,40),LMN(40,40,3),NLMN(40,40)
5.   COMMON/CONS/NBTNEU,NBTPAR,NBOUC
6.   DIMENSION TPBUS(12,12,14),TMP(40,40),TI(14)
7.   DIMENSION TP(14),TS(14),LDMAN(14)
8.   INTEGRER TPBUS,TMP
9. C INITIALISATION
10.  GIGA=1.E+20
11.  TTMIN=GIGA
12.  TIM=GIGA
13. C INDICATEUR DE POSSIBILITES
14.  ICAS(M,N)=3
15.  DO 1L=1,NBTPAR
16.  I=NOSTA(M,L)
17.  J=NOSTA(N,L)
18.  IJ=I*J
19.  IF(IJ.EQ.0.OR.I.GE.J) GO TO 1
20.  U=FLOAT(TPBUS(I,J,L))
21.  IF(U.GE.GIGA.OR.U.GT.FLOAT(TMP(M,N))) GO TO 1
22.  ICAS(M,N)=2
23.  IF(U+TI(L).LT.TTMIN)TTMIN=U+TI(L)
24.  1 CONTINUE
25.  IF(ICAS(M,N).EQ.3) RETURN
26.  IF(TTMIN.LE.FLOAT(TMP(M,N))) ICAS(M,N)=1
27.  DF=AMIN1(TTMIN,FLOAT(TMP(M,N)))
28.  NC=0
29.  DO 2L=1,NBTPAR
30.  I=NOSTA(M,L)
31.  J=NOSTA(N,L)
32.  IJ=I*J
33.  IF(IJ.EQ.0.OR.I.GE.J) GO TO 2
34.  TP(L)=FLOAT(TPBUS(I,J,L))
35.  IF(TP(L).GE.DF) GO TO 2
36.  NC=NC+1
37.  TS(NC)=TP(L)+TI(L)
38.  IF(TI(L).LT.TIM) TIM=TI(L)
39.  LDMAN(NC)=L
40.  2 CONTINUE
41.  NLMN(M,N)=NC
42.  IF(NC.GT.NLMAX) GO TO 8
43.  DO 9K=1,NC
44.  9 LMN(M,N,K)=LDMAN(K)
45.  RETURN
46. C RANGEMENT PAR ORDRE CROISSANT
47.  8 NCL=NC+1
```

```
48.      DO 4I=1,NC1
49.      I1=I+1
50.      DO 5J=I1,NC
51.      IF(TS(I).LT.TS(J)) GO TO 5
52.      T=TS(J)
53.      TS(J)=TS(I)
54.      TS(I)=T
55.      L=LDMAN(J)
56.      LDMAN(J)=LDMAN(I)
57.      LDMAN(I)=L
58.      5 CONTINUE
59.      IF(I.EQ.NLMAX) GO TO 3
60.      4 CONTINUE
61.      3 TIM=GIGA
62.      NLMN(M,N)=NLMAX
63.      DO 6K=1,NLMAX
64.      LMN(M,N,K)=LDMAN(K)
65.      IF(TI(K).LT.TIM) TIM=TI(K)
66.      6 CONTINUE
67.      RETURN
68.      END
```

```
1. C AFFECTATION DES USAGERS
2.   SUBROUTINE AFFECT(NBMSTA,NBMPPB,TTM,IOD,PBUS,TRANCH,
3.   2TIMAX,TPPT,NBTBUS,IW)
4.   COMMON/DESC/NEULI(7,14),NBSTA(14),NOSTA(40,14)
5.   COMMON/COMP/ICAS(40,40),LMN(40,40,3),NLMN(40,40)
6.   COMMON/CONS/NBTNEU,NBTPAR,NBOUC
7.   DIMENSION TTM(40,40),IOD(40,40),PBUS(40,40),PROB(40,
8.   240,3),KCOR(40,40),TIMAX(9),NBPPST(8,14),TPPT(9),PAS
   MAX(9)
9.   INTEGER TPPT,TRANCH
10.  CALL CORGPO(TTM,KCOR)
11.  IW=1
12.  IF(IW.NE.1) GO TO 30
13.  WRITE(IO8,2I4)
14.  WRITE(IO8,50)((KCOR(M,N),N=1,NBTNEU),M=1,NBTNEU)
15.  WRITE(IO8,400)
16.  WRITE(IO8,40I)((TTM(M,N),N=1,NBTNEU),M=1,NBTNEU)
17.  30 CONTINUE
18.  DO 35L=1,NBTPAR
19.  DO 35I=1,NBMSTA-1
20.  35 NBPPST(I,L)=0
21.  C CALCUL DU CRITERE OBJECTIF
22.  TTPA=0.
23.  DO 1M=1,NBTNEU
24.  DO 2N=1,NBTNEU
25.  IF(M.EQ.N) GO TO 2
26.  TTPA=TTPA+FLOAT(IOD(M,N))*TTM(M,N)
27.  MO=M
28.  6 ND=KCOR(MO,N)
29.  IF(ICAS(MO,ND).EQ.3) GO TO 7
30.  NLL=NLMN(MO,ND)
31.  FR=PBUS(MO,ND)
32.  DO 3IP=1,NLL
33.  L=LMN(MO,ND,IP)
34.  PAS=IFIX(FLOAT(IOD(M,N))*FR*PROB(MO,ND,IP))+1
35.  I=NOSTA(MO,L)
36.  J=NOSTA(ND,L)-1
37.  IF(J.EQ.0)J=NBSTA(L)-1
38.  DO 4K=1,J
39.  NBPPST(K,L)=NBPPST(K,L)+PAS
40.  4 CONTINUE
41.  3 CONTINUE
42.  7 IF(ND.EQ.N) GO TO 2
43.  MO=ND
44.  GO TO 6
45.  2 CONTINUE
46.  1 CONTINUE
47.  WRITE(IO8,300)
48.  WRITE(IO8,98)TTPA
49.  IW=1
50.  IF(IW.NE.1)GO TO 19
51.  DO 25LX=1,NBTPAR
52.  KOK=NBSTA(LX)-1
53.  WRITE(IO8,2I5)
54.  25 WRITE(IO8,IO6)((NBPPST(K,LX),K=1,KOK),LX=1,NBTPAR)
55.  19 CONTINUE
56.  NL=1
57.  NDEP=NEULI(1,1)
```

```
58.      PGC=0
59.      DO 8LJ=1,NBTPAR
60.      NP=NBSTA(LJ)-1
61.      DO 10 II=1,NP
62.      IF(PGC.LT.NBPPST(II,LJ))PGC=NBPPST(II,LJ)
63.      IO CONTINUE
64.      IF(NDEP.NE.NEULI(NBSTA(LJ),LJ)) GO TO 8
65.      C CALCUL DES CHARGEMENTS AUX INTERSTATIONS
66.      PSMAX(NL)=PGC
67.      TIMAX(NL)=FLOAT(TRANCH)*FLOAT(NBMPPB)/FLOAT(PASMAX(NL))
68.      NL=NL+1
69.      NDEP=NEULI(1,LJ+1)
70.      8 CONTINUE
71.      C CALCUL DU NOMBRE DE BUS NECESSAIRE
72.      NBNEC=0
73.      DO 15NL=1,NBOUC
74.      NBNEC=NBNEC+IFIX(FLOAT(TPTT(NL))/TIMAX(NL))+1
75.      I5 CONTINUE
76.      IF(NBNEC-NBTBUS)16,16,17
77.      I6 WRITE(I08,I22)
78.      GO TO 36
79.      I7 WRITE(I08,I23)
80.      36 WRITE(I08,2I6)
81.      WRITE(I08,I24)NBNEC
82.      IW=1
83.      IF(IW.NE.1)RETURN
84.      WRITE(I08,2I7)
85.      WRITE(I08,I26)(TIMAX(NL),NL=1,NBOUC)
86.      WRITE(I08,2I8)
87.      WRITE(I08,I25)(PSMAX(NL),NL=1,NBOUC)
88.      50 FORMAT(40 I3)
89.      98 FORMAT(10X,F10.2)
90.      I06 FORMAT(14I8)
91.      I22 FORMAT(25H LE RESEAU EST ACCEPTABLE)
92.      I23 FORMAT(31H LE RESEAU N'EST PAS ACCEPTABLE)
93.      I24 FORMAT(1X,I3)
94.      I25 FORMAT(9F7.2)
95.      I26 FORMAT(1X,9I6)
96.      2I4 FORMAT('TABLEAU KCOR(M,N)')
97.      2I5 FORMAT('CHARGEMENTS AUX INTERSTATIONS NBPPST(I,L)')
98.      2I6 FORMAT('NOMBRE DE BUS NECESSAIRE')
99.      2I7 FORMAT('CHARGEMENT MAXIMUM PSMAX(NL)')
I00.      2I8 FORMAT('TEMPS D'INTERVALLE TIMAX(NL)')
I01.      300 FORMAT('VALEUR DE LA FONCTION OBJECTIF')
I02.      400 FORMAT(24H TEMPS DE TRAJETS MOYENS)
I03.      40I FORMAT(20F6.2)
I04.      RETURN
I05.      END
```

```
1.  C RECHERCHE DES NOEUDS DE CORRESPONDANCE
2.  SUEROUTINE COREPO(TTM, KCOR)
3.  COMMON/CONS/NBTNEU, NBT PAR, NBOUC
4.  DIMENSION TTM(40,40), KCOR(40,40)
5.  DO 5 M=1, NBTNEU
6.  DO 5 N=1, NBTNEU
7.  5 KCOR(M,N)=N
8.  DO 10 K=1, NBTNEU
9.  DO 20 M=1, NBTNEU
10. IF(M.EQ.K) GO TO 20
11. DO 30 N=1, NBTNEU
12. S=TTM(M,K) + TTM(K,N)
13. IF(TTM(M,N).LT.S) GO TO 30
14. TTM(M,N)=S
15. KCOR(M,N)=KCOR(M,K)
16. 30 CONTINUE
17. 20 CONTINUE
18. 10 CONTINUE
19. RETURN
20. END
```

```
1.      C CALCUL DES COEFFICIENTS DU POLYNOME
2.          FUNCTION COFPOL(KP,N,T)
3.          DIMENSION T(I),IRAN(I4)
4.          CK=0.
5.          DO I0 KX=I,KP
6.      I0  IRAN(KX)=0.
7.          KX=I
8.      II  IRAN(KX)=IRAN(KX) + I
9.          IF(KX.EQ.KP) GO TO I3
10.     KXI=KX + I
11.     DO I2 KXX=KXI,KP
12.     I2  IRAN(KXX)=IRAN(KXX - I) + I
13.     I3  APK=I.
14.     DO I4 KXX=I,KP
15.     I4  APK=APK/T(IRAN(KXX))
16.     CK=CK + APK
17.     IF(IRAN(KX).GE.N-KP+KX) GO TO I5
18.     KX=KP
19.     GO TO II
20.     I5  KX=KX - I
21.     IF(KX.GE.I) GO TO II
22.     COFPOL=CK
23.     RETURN
24.     END
```



III. Analyse et interprétation des résultats :

Nous avons supposé qu'on avait le même nombre d'autobus sur chacune des boucles du réseau.

Dans le programme principal, nous calculons le nombre d'autobus utilisé, ainsi que le nombre de véhicules affectés sur chaque boucle, soit 5 dans notre exemple.

La matrice NOSTA (I,L) , indiquant le rang de chaque noeud sur tous les parcours, est donnée sur le tableau n°8.

Les temps de rotation sur les différentes boucles sont les suivants:

NL	I	2	3	4	5	6	7	8	9
TPTT (NL)	70	84	90	123	87	57	57	43	50

Tableau n°9.

Les temps d'intervalles par boucle sont:

NL	X (NL)
I	14.00
2	16.80
3	18.00
4	24.60
5	17.40
6	11.40
7	11.40
8	8.60
9	10.00

Tableau n°10.

TABLEAU NOSTA ( I , L )

Tableau n° 8.

	I	2	3	4	5	6	7	8	9	10	11	12	13	14
I	I	4	I	6	I	7	I	6	I	7	I	I	0	0
2	0	3	0	0	0	0	0	0	0	0	0	0	0	0
3	0	2	0	0	0	0	0	0	0	0	0	0	0	0
4	4	I	0	0	0	0	0	0	0	0	0	0	0	0
5	3	0	0	0	0	0	0	0	0	0	0	0	I	I
6	2	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	2	0	2	0	2	0	0	0	0	0	0	0
8	0	0	3	0	3	0	3	0	0	0	0	0	0	0
9	0	0	0	3	0	0	0	0	0	0	0	0	0	0
10	0	0	0	2	0	0	0	0	0	0	0	0	0	0
11	0	0	6	I	0	0	0	0	0	0	0	0	0	0
12	0	0	5	0	5	0	0	0	0	0	0	0	0	0
13	0	0	4	0	4	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	4	2
15	0	0	0	0	0	0	0	0	0	0	0	0	3	3
16	0	0	0	0	0	0	0	0	0	0	0	0	2	4
17	0	0	0	5	0	6	0	5	0	0	0	0	0	0
18	0	0	0	4	0	5	0	4	0	0	0	0	0	0
19	0	0	0	0	0	4	0	0	0	0	0	0	0	0
20	0	0	0	0	0	3	0	0	0	0	0	0	0	0
21	0	0	0	0	0	2	0	0	0	0	0	0	0	0
22	0	0	0	0	7	I	0	0	0	0	0	0	0	0
23	0	0	0	0	6	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	4	0	0	0	0	0	0	0
25	0	0	0	0	0	0	5	0	0	0	0	0	0	0
26	0	0	0	0	0	0	6	I	0	0	0	0	0	0
27	0	0	0	0	0	0	0	2	0	0	0	0	0	0
28	0	0	0	0	0	0	0	3	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	2	0	0	0	0	0
30	0	0	0	0	0	0	0	0	3	0	0	0	0	0
31	00	0	0	0	0	0	0	0	4	0	0	0	0	0
32	0	0	0	0	0	0	0	0	5	I	0	0	0	0
33	0	0	0	0	0	0	0	0	0	2	0	0	0	0
34	0	0	0	0	0	0	0	0	0	3	0	0	0	0
35	0	0	0	0	0	0	0	0	0	4	0	0	0	0
36	0	0	0	0	0	0	0	0	0	5	5	3	0	0
37	0	0	0	0	0	0	0	0	0	6	6	2	0	0
38	0	0	0	0	0	0	0	0	0	0	3	5	0	0
39	0	0	0	0	0	0	0	0	0	0	4	4	0	0
40	0	0	0	0	0	0	0	0	0	0	2	6	0	0

Les temps d'intervalles sur chaque parcours sont:

L	TI (L)
I	I4.00
2	I4.00
3	I6.80
4	I6.80
5	I8.00
6	I8.00
7	24.60
8	24.60
9	I7.40
I0	I7.40
II	II.40
I2	II.40
I3	8.60
I4	I0.00

Tableau n°II.

Des difficultés rencontrées en cours de programmation nous ont amené à faire la suite des calculs "manuellement".

Nous donnons sur le tableau n°I2 l'indicateur de possibilités ICAS (M,N) pour chaque trajet direct M-N.

- ICAS (M,N) = I : M-N est un trajet-bus
- ICAS (M,N) = 2 : M-N est un trajet mixte
- ICAS (M,N) = 3 : M-N est:
  - ou bien un trajet pied
  - ou bien un trajet "non-direct"

Le nombre et les numéros des parcours concurrents sur chaque trajet direct sont fournis par les tableaux I3 et I4 respectivement.

# Indicateur de possibilités ICAS (M,N)

M \ N	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	27	28	29	30	31	32	33	34	35	36	37	38	39	40	
1	0	3	3	3	3	2	3	2	3	3	1	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	2	3	3	3	2	2	3	2	2
2	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
4	2	2	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
5	3	3	3	3	0	3	3	3	3	3	3	3	3	3	2	1	1	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
6	3	3	3	2	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
7	3	3	3	3	3	0	2	3	3	2	1	1	3	3	3	3	3	3	3	3	3	3	3	2	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3
8	3	3	3	3	3	3	0	3	3	2	2	2	3	3	3	3	3	3	3	3	3	3	2	1	3	3	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3
9	2	3	3	3	3	3	0	3	3	3	3	3	3	3	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
10	2	3	3	3	3	3	2	0	3	3	3	3	3	3	1	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
11	2	3	3	3	3	3	3	2	2	0	3	3	3	3	1	1	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
12	3	3	3	3	3	3	3	3	3	3	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
13	3	3	3	3	3	3	3	3	3	2	2	0	3	3	3	3	3	3	3	3	3	3	3	3	1	1	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
14	3	3	3	3	1	3	3	3	3	3	3	3	3	3	0	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
15	3	3	3	3	1	3	3	3	3	3	3	3	3	3	2	0	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
16	3	3	3	3	1	3	3	3	3	3	3	3	3	3	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
17	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
18	2	3	3	3	3	3	3	3	3	3	3	3	3	3	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
19	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
20	2	3	3	3	3	3	3	3	3	3	3	3	3	3	2	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
21	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
22	1	3	3	3	3	3	3	3	3	3	3	3	3	3	1	1	2	2	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
23	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
24	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
25	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
26	1	3	3	3	3	3	3	3	3	3	3	3	3	3	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
27	2	3	3	3	3	3	3	3	3	3	3	3	3	3	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
28	2	3	3	3	3	3	3	3	3	3	3	3	3	3	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
29	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
30	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
31	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
32	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
33	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
34	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
35	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
36	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
37	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
38	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
39	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
40	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3

Tableau n° 12

# Nombre de parcours concurrents

M \ N	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	27	28	29	30	31	32	33	34	35	36	37	38	39	40					
1	0	0	0	0	0	1	0	3	0	0	1	2	2	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	1	1	1	1	0	0	0	1	1	0	1	1				
2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
4	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
5	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
6	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
7	0	0	0	0	0	0	3	0	0	1	2	2	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
8	0	0	0	0	0	0	0	0	0	1	2	2	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
9	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
10	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
11	1	0	0	0	0	0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
13	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
14	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
15	0	0	0	0	2	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
16	0	0	0	0	2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
18	3	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
19	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
20	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
21	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
22	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
26	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0				
27	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
28	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0				
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0				
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
32	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0				
33	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0			
34	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0		
35	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0		
36	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	1	0	
37	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
39	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	
40	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	

Tableau n° 13



Le tableau n°I5 indique la proportion de passagers sur chaque parcours concurrent.

Le tableau n°I6 renseigne sur la proportion d'usagers qui prennent le bus sur chaque trajet direct.

En utilisant les temps de marche à pied, les temps de trajet en autobus sur les trajets qui ne sont pas des trajets pied, nous avons établi la matrice des noeuds de correspondance KCOR (M,N). Cette matrice indique le premier noeud de correspondance sur chaque trajet.

ex:  $\text{KCOR}(1,8) = 8$  : il n'y a pas de noeud de correspondance entre les noeuds 1 et 8.

$\text{KCOR}(2,37) = 1$  : le premier noeud de correspondance entre les noeuds 2 et 37 est le noeud n°1.

$\text{KCOR}(4,7) = 2$

$\text{KCOR}(2,7) = 6$

$\text{KCOR}(6,7) = 7$  ; le trajet 4-7 comporte 2 noeuds de correspondance: les noeuds 2 et 6.

La matrice KCOR (M,N) est figurée sur le tableau n°I8.

A partir de cette matrice, nous avons déduit la matrice des temps de trajet moyens TTM (M,N).

Vu que les valeurs numériques des temps de marche à pied ne reposent sur aucune réalité physique, nous avons constaté que, parfois, pour un trajet pied, il existait un ou plusieurs noeuds de correspondance tels que le trajet s'effectue quand-même à pied. Il est alors nécessaire de refaire une simulation avec les nouvelles valeurs des temps de marche à pied. Il est à noter que ce problème ne se pose pas lors de la simulation d'un réseau réel.

Nous allons voir sur un exemple comment, avec le modèle de simulation choisi, nous sommes arrivés à calculer tous les temps de trajet moyens, et par suite, la fonction objectif.









# Tableau KCOR(M,N)

	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	27	28	29	30	31	32	33	34	35	36	37	38	39	40	
1	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	27	28	29	30	31	32	33	34	35	36	37	38	39	40	
2	1	2	3	4	5	6	6	6	6	6	9	9	9	9	7	7	9	19	20	13	12	8	24	25	24	8	7	7	29	28	29	1	1	36	1	1	36	1	1	36	1
3	1	2	3	5	5	2	6	5	5	5	10	9	14	15	5	2	5	9	19	20	12	10	19	24	25	25	19	7	7	28	28	29	1	1	2	2	2	2	36	2	
4	1	2	3	4	5	2	2	5	14	10	10	10	9	14	15	16	7	9	9	19	20	13	12	19	24	25	25	19	7	28	28	29	32	31	31	1	1	1	1	36	3
5	1	2	3	4	5	6	7	8	9	10	11	10	9	14	15	16	3	18	9	19	20	12	10	19	24	25	24	19	17	29	28	28	29	1	1	2	1	1	36	1	
6	1	2	2	4	5	6	7	7	9	10	7	7	9	9	9	9	11	7	9	19	20	13	20	8	24	25	24	8	17	29	29	28	29	1	1	2	2	2	2	36	2
7	1	6	1	3	1	6	7	8	8	12	12	13	9	9	8	17	18	19	19	21	22	23	24	25	26	24	8	17	29	29	28	29	33	35	28	1	1	39	40		
8	1	6	6	5	5	6	18	8	9	10	11	12	13	9	9	9	17	18	19	19	20	22	23	24	25	24	24	28	29	29	28	29	31	31	28	37	29	39	4		
9	1	6	5	1	5	6	8	8	9	10	13	10	13	14	15	16	17	8	19	19	20	13	20	19	24	24	24	24	8	8	28	28	33	31	31	28	9	29	8	7	
10	1	9	5	16	5	9	9	9	9	10	12	12	13	16	15	16	17	18	18	12	20	12	23	19	24	24	24	13	8	28	28	29	32	31	31	28	8	29	28	7	
11	1	10	5	10	5	5	9	8	9	10	11	12	10	14	10	10	17	18	12	12	20	12	14	19	24	24	24	19	8	28	29	29	32	31	28	28	8	29	28	8	
12	10	10	5	10	5	10	10	10	9	10	11	12	10	10	10	8	8	9	20	20	22	23	19	24	24	24	19	8	28	28	29	32	31	28	28	9	30	29	7		
13	17	9	9	10	9	9	9	9	10	11	12	13	9	10	10	17	8	19	19	20	22	23	24	25	24	24	28	8	28	28	29	32	31	28	28	8	29	34	7		
14	4	9	5	4	5	5	9	9	9	10	11	10	9	14	15	16	8	9	9	19	20	13	20	19	24	24	9	19	8	28	28	32	31	28	8	8	8	8	7		
15	9	10	5	14	5	5	9	9	9	10	10	10	10	14	15	16	8	9	9	12	20	12	10	9	24	24	9	19	8	28	28	32	31	28	28	8	8	8	7		
16	10	10	5	5	5	10	10	9	9	10	10	10	10	14	15	16	13	9	10	12	20	12	10	19	24	9	13	13	8	28	29	27	32	31	28	28	8	8	8	7	
17	7	7	7	7	7	7	7	8	9	10	9	9	9	8	13	17	7	9	19	20	13	20	8	24	24	25	28	29	29	28	29	33	31	7	1	29	7	7			
18	1	8	8	8	5	7	7	8	8	9	10	10	9	9	8	13	17	18	8	19	20	13	8	24	24	24	24	28	8	28	28	32	31	28	28	8	8	8	7		
19	1	9	9	13	9	7	7	8	9	9	13	20	13	9	9	13	17	18	19	20	20	13	20	24	24	24	24	28	8	28	28	32	31	28	28	8	8	28	7		
20	1	19	19	12	19	2	19	19	19	9	12	12	9	10	12	17	18	19	20	21	22	23	19	25	24	25	19	8	7	28	28	32	31	28	28	8	8	28	7		
21	1	20	20	20	20	20	20	20	20	20	20	20	9	9	10	20	17	18	19	20	21	22	23	19	25	24	27	28	8	28	28	32	32	32	28	28	8	30	28	7	
22	1	21	21	12	21	20	21	21	21	21	21	21	9	10	21	17	18	19	20	21	22	23	24	25	26	25	21	8	28	28	28	32	32	28	28	8	30	28	7		
23	20	20	20	10	10	20	19	20	10	10	10	10	10	10	10	8	8	19	20	21	22	23	19	25	26	25	21	8	28	28	28	32	32	28	28	8	8	28	8		
24	18	20	8	13	19	19	8	8	9	9	13	10	13	9	8	13	8	18	19	19	23	21	23	24	25	25	27	28	8	28	28	28	32	31	28	28	8	30	28	8	
25	24	24	24	24	24	24	24	24	24	24	9	24	9	8	24	24	8	24	24	21	22	23	24	25	26	27	24	8	27	27	31	27	31	27	27	8	30	35	7		
26	1	25	25	25	25	25	25	25	25	25	10	10	25	9	10	10	17	18	20	19	21	22	22	24	25	26	27	28	8	28	32	32	32	34	34	35	36	30	35	37	
27	1	25	25	25	25	25	25	25	9	10	9	9	8	25	25	18	25	21	21	22	23	24	25	26	27	28	8	30	31	31	33	34	35	36	35	30	35	38			
28	1	8	8	24	24	2	17	8	19	9	10	13	13	9	8	13	17	18	19	19	21	21	21	24	24	24	28	8	30	31	32	32	31	35	36	8	30	39	7		
29	17	17	17	17	17	2	17	8	8	9	10	9	9	9	8	13	17	8	8	19	20	21	21	8	24	24	27	8	29	30	31	32	33	33	31	7	37	39	30	7	
30	29	29	29	29	29	2	29	29	29	9	29	9	28	9	8	13	29	8	28	19	20	22	20	28	25	24	27	28	29	30	31	32	34	31	34	35	37	37	39	38	
31	29	29	28	29	28	2	29	28	28	9	28	10	28	28	28	13	28	9	13	28	21	21	21	28	25	24	27	28	29	30	31	32	33	34	35	34	35	30	35	37	
32	1	31	31	28	28	2	7	8	8	9	10	10	28	9	10	10	31	8	13	28	21	22	21	28	24	26	27	28	8	30	31	32	33	34	35	36	37	37	35	37	
33	1	1	31	32	32	1	7	32	28	32	10	10	28	9	10	32	7	8	13	28	21	22	21	28	24	26	27	28	8	30	31	32	33	34	35	36	37	37	35	40	
34	1	1	1	31	31	1	7	8	9	32	10	10	28	9	9	10	7	8	9	28	21	22	21	28	24	24	28	17	30	31	32	33	34	35	36	37	37	35	36		
35	1	1	1	4	1	1	7	7	28	31	10	10	28	9	8	13	1	7	13	28	20	21	20	28	24	34	27	28	7	31	31	32	33	34	35	36	37	37	39	37	
36	1	1	1	2	1	2	7	8	28	9	10	10	28	7	8	13	1	7	8	28	20	21	21	28	24	35	27	28	7	7	31	32	33	34	35	36	37	37	39	37	
37	1	1	1	1	1	1	7	8	7	10	7	9	9	8	13	1	7	8	19	20	21	20	8	24	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40		
38	1	1	1	1	1	1	7	29	9	10	10	9	9	9	13	1	7	9	19	20	22	20	8	24	24	25	28	29	30	30	30	33	34	35	36	37	38	39	40		
39	1	36	1	2	2	2	7	8	8	9	10	9	9	9	8	13	7	8	8	28	20	21	20	24	24	24	24	28	29	30	31	32	32	34	35	36	37	38	39	40	
40	1	1	1	1	1	1	7	7	7	7	8	7	9	9	8	13	7	7	8	19	7	17	7	8	24	24	24	8	7	30	29	32	33	34	35	36	37	38	39	40	

Tableau n° 18

exemple: il s'agit du trajet 36-37 :

Nous voyons sur la figure I que 2 lignes passent par les noeuds 36 et 37 dans le sens 36 vers 37, ce sont les lignes IO et II.

Le temps de marche à pied pour aller de 36 à 37 est  $\tau ( 36 , 37 ) = 10$  mn

Sur les parcours IO et II, les noeuds 36 et 37 sont respectivement la 5-ième et la 6-ième station.

Le temps de trajet en autobus, supposé le même pour les deux parcours, est:

$$\theta ( 5, 6, IO ) = \theta ( 5, 6, II ) = 7 \text{ mn.}$$

Les temps d'intervalles sont :

$$TI ( IO ) = 17.40 \text{ mn}$$

$$TI ( II ) = 11.40 \text{ mn}$$

$$\text{Min} \left[ \theta ( 5, 6, IO ) + TI ( IO ), \theta ( 5, 6, II ) + TI ( II ) \right] = 18.40 \text{ mn}$$

nous constatons que :

$$7 < \tau ( 36, 37 ) < 18.40$$

il s'agit donc d'un trajet mixte,  $I_p ( 36, 37 ) = 2$

Nous allons voir maintenant quel est le pourcentage de gens qui prendront le bus sur ce trajet direct. Nous utilisons pour cela la relation donnée dans le modèle.

$$R_A ( 36, 37 ) = 1 - \left[ 1 - \frac{10 - 7}{17.4} \right] \left[ 1 - \frac{10 - 7}{11.4} \right]$$

$$R_A ( 36, 37 ) = 0.39019$$

Ainsi 39% des gens prendront le bus, le reste, soit 61%, partiront à pied.

Les gens qui prennent le bus seront véhiculés par les autobus affectés aux lignes IO et II.

Le calcul des différentes proportions de passagers sur chacune des lignes concurrentes est effectué en utilisant la relation:

$$R_p(m,n,l) = \int_0^{T_{Imin}} \frac{I}{T_I(p_k)} \cdot \prod_{\substack{i=I \\ i \neq k}}^{L(m,n)} \left[ I - \frac{t}{T_I(i)} \right] dt$$

Le calcul donne les résultats suivants:

$$R_p(36,37,1) = 0.32$$

$$R_p(36,37,2) = 0.68$$

Finalement, parmi les gens qui se présentent au nœud 36 pour se rendre au nœud 37;

12.4% seront véhiculés par les bus de la ligne IO,

26.6% par les bus de la ligne II,

61% iront à pied.

Calcul du temps de trajet moyen:

Pour ce faire, il reste à calculer le temps moyen d'attente.  $E(z)$ .

$$E(z) = \int_0^3 t \cdot dF_R(t)$$

$$F_R(t) = I - \left( I - \frac{t}{I7.4} \right) \cdot \left( I - \frac{t}{II.4} \right)$$

$$= \frac{t}{I7.4} + \frac{t}{II.4} - \frac{t^2}{I92.36}$$

$$dF_R(t) = 0.145 - \frac{t}{99.18}$$

$$E(z) = \int_0^3 \left( 0.145t - \frac{t^2}{99.18} \right) dt$$

$$E(z) = 0.59$$

$$\text{d'où : } \bar{t}(36,37) = 0.59 + (0.39 \times 7) + (0.61 \times 10)$$

$$\bar{t}(36,37) = 9.42 \text{ mn}$$

calcul de la fonction objectif:

Nous donnons dans le tableau suivant le détail des calculs de la fonction objectif.

$$\text{Nous savons que } TTPA = \sum_{\substack{m=1 \\ n=1}}^{40} OD(m,n) \cdot \bar{t}(m,n)$$

Nous indiquons les valeurs des produits  $OD(m,n) \cdot \bar{t}(m,n)$  pour variant de 1 à 40.

n

n	$\sum_{m=1}^{40} OD(m,n) \cdot \bar{t}(m,n)$	n	$\sum_{m=1}^{40} OD(m,n) \cdot \bar{t}(m,n)$
I	49 459.82	21	66 162.59
2	64 183.11	22	87 680.01
3	72 812.71	23	77 303.00
4	97 954.32	24	49 146.83
5	73 004.41	25	73 122.63
6	59 688.93	26	102 528.64
7	57 209.01	27	59 719.78
8	51 707.31	28	48 361.24
9	52 757.03	29	58 740.29
10	68 972.88	30	61 171.06
11	97 565.63	31	60 773.45
12	80 173.21	32	65 623.73
13	55 118.30	33	68 162.61
14	68 849.73	34	67 092.99
15	81 501.76	35	63 643.48
16	72 779.79	36	65 857.07
17	58 924.63	37	63 178.92
18	55 577.06	38	63 060.47
19	52 572.64	39	58 528.09
20	60 295.93	40	53 226.12

Tableau n°19

La valeur finale de la fonction objectif est:

$$TTPA = 2\ 644\ 191.21\ \text{mn}$$

soit 44 069 h 51 mn 12 s.

Calcul de PASMAL (NL) :

Pour toutes les interstations composant une boucle, on calcule les différents chargements. On procède de la façon suivante:

Pour un couple de noeuds donné M et N, on calcule le nombre de passagers ayant emprunté le bus par voie directe (trajet direct.), ou ayant pris la correspondance sur cette interstation, et ce, en utilisant la relation:

$$PAS = OD ( M , N ) \times R_A ( M \emptyset , ND ) \times R_p ( M \emptyset , ND , IP )$$

Pour chaque interstation on additionne les résultats trouvés, la somme obtenue correspond au chargement total sur l'interstation considérée. Pour chaque boucle, on compare les différents chargements aux interstations, on prend ensuite pour PASMAY la plus grande des valeurs trouvées.

La matrice des chargements aux interstations sur chaque parcours est représentée sur le tableau n°20.

Les plus grands chargements sur chaque boucle se déduisent aisément à partir de ce tableau.

Nous rappelons que les temps d'intervalles maximaux sur les différentes boucles sont calculés en utilisant la relation :

$$TIMAX (NL) = TRANCH \times NBMPPB / PASMAY (NL)$$

nous avons pris ; TRANCH = 120

NBMPPB = 100

Le tableau n°21 contient les valeurs de PASMAY et TIMAX pour chaque boucle:

#### Calcul du nombre d'autobus nécessaire:

Nous rappelons que l'on entend par nombre de bus nécessaire le nombre de véhicules nécessaire pour que tout le monde soit transporté normalement, c'est à dire pour que personne ne resten "rade".



interstation P	I	2	3	4	5	6
I	49	3I	23	0	0	0
2	I83	204	I3I	0	0	0
3	422	20I7	544	632	3I2	0
4	I99	435	2I7	220	I25	0
5	3I9	I9I3	7I5	I244	I426	I040
6	375	449	I302	5I5	342	I20
7	2I9	I065	II9	89	I088	0
8	I47	I77	I66	I70	I42	0
9	44	36I	750	277	0	0
I0	I63	454	767	I378	I238	II20
II	73	I9	0	I00	290	767
I2	I98	I35	67	I23	I80	405
I3	230	794	762	84I	0	0
I4	308	2II	I65	550	0	0

Tableau n° 20.

NL	PASMAX (NL)	TIMAX (NL)
I	204	58.82
2	20I7	5.94
3	I9I3	6.27
4	I088	II.02
5	I378	8.70
6	767	I5.64
7	405	29.62
8	84I	I4.26
9	550	2I.8I

Tableau n° 2I.

$$NBNEC = \sum_{NL=1}^9 \frac{TPTT (NL)}{TIMAX (NL)}$$

NL	1	2	3	4	5	6	7	8	9
$\frac{TPTT (NL)}{TIMAX (NL)}$	2	15	15	12	10	4	2	4	3

d'où: NBNEC = 67

L'affectation proposée des autobus sur les différentes boucles n'est pas acceptable car la compagnie a été supposée ne disposer que de 50.

Ce résultat était à prévoir car il est évident qu'on affecte pas le même nombre d'autobus sur des boucles différentes du point de vue importance du trafic.

IV. CONCLUSION :

-----

Nous avons montré à travers cet exemple une application du modèle de simulation proposé. Il est évident que ceci ne constitue que le premier pas vers une étude d'optimisation.

Le programme de simulation proposé n'est pas tout à fait au point, il est à améliorer.

Pour l'optimisation, il reste à écrire un sous-programme qui calculera le gradient de la fonction objectif. La recherche du réseau optimal se fait alors automatiquement, elle nécessite cependant un grand nombre de simulations.

