



THESIS / THÈSE

MASTER EN SCIENCES MATHÉMATIQUES

Les algorithmes directs en programmation complémentaire

Malherbe, Anne-Marie

Award date:
1978

Awarding institution:
Universite de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LES ALGORITHMES
DIRECTS
EN PROGRAMMATION
COMPLEMENTAIRE

MALHERBE

ANNE - MARIE

FMB1/1978/13

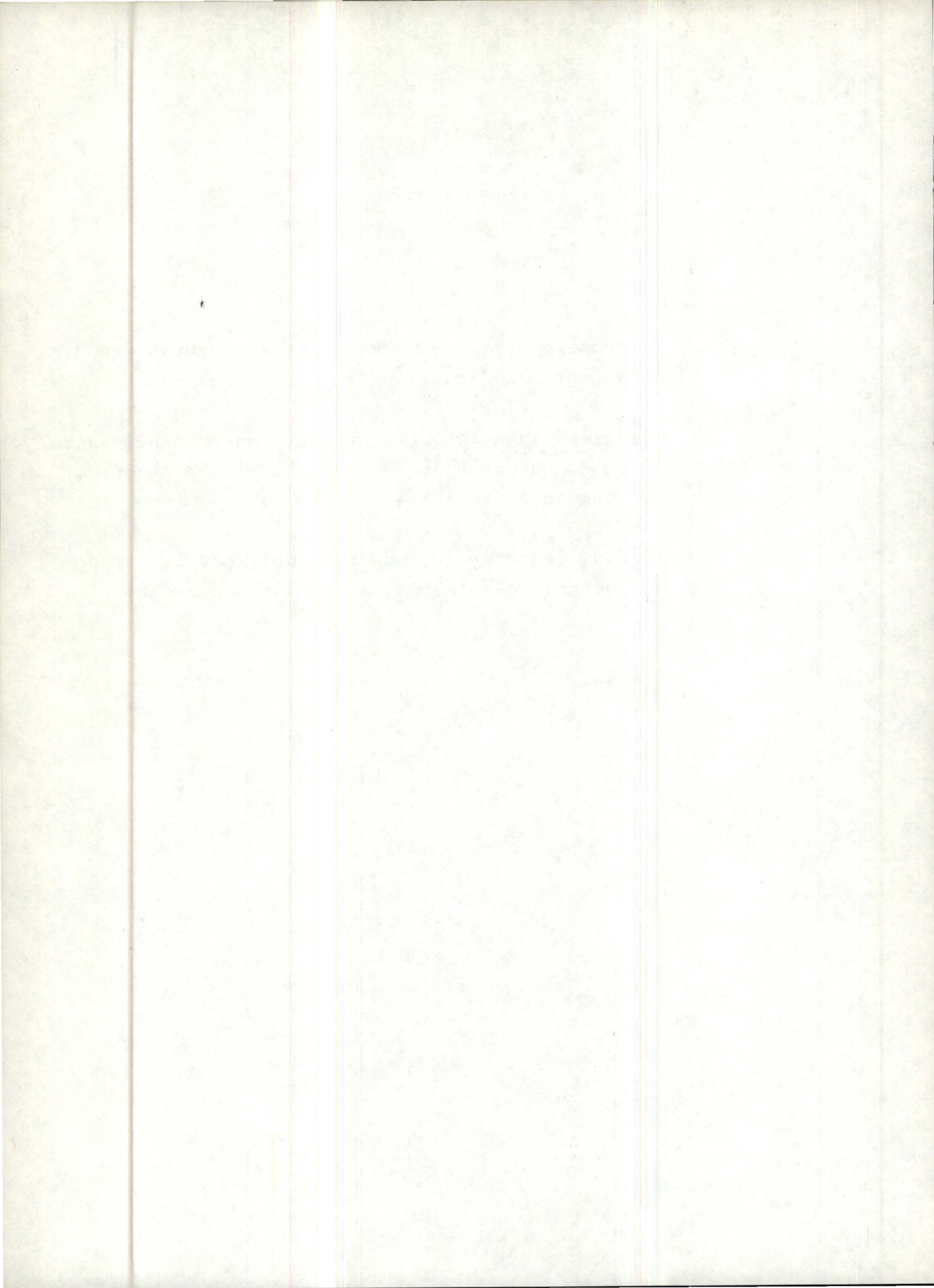


6520 - 13242

Je remercie bien vivement Monsieur Nguyen van Hien d'avoir accepté la direction de ce mémoire.

Je tiens aussi à exprimer toute ma reconnaissance à Monsieur C. Duchateau pour l'aide efficace apportée tout au long de cette année.

Enfin, je remercie tous ceux qui de près ou de loin ont contribué à la réalisation de ce mémoire.



I N T R O D U C T I O N

- - - - -
- - - - -

Les programmations linéaire et quadratique, la théorie des jeux bimatriciels, et bien d'autres applications mathématiques, amènent à considérer des problèmes de complémentarité. La recherche d'une méthode efficace de résolution de ces problèmes s'impose donc.

Lemke, un des premiers, a mis au point une méthode de résolution systématique de ces problèmes. Aussi, dans le premier chapitre de ce mémoire, après avoir défini les notions indispensables à notre étude, nous analyserons son algorithme comme technique de résolution [15].

D'autres, comme Cottle et Dantzig, proposent une méthode de pivotage principal dont nous exposerons brièvement la "philosophie" [5], [6].

Procédant du même esprit, viennent ensuite les méthodes "directes" qui utilisent les données même du problème sans introduire de variables artificielles, et dont les règles d'exécution sont directes, concises et simples à appliquer.

Dans le chapitre II, nous introduisons une propriété (la plus fructueuse sans doute dans ce mémoire) des solutions des problèmes de complémentarité liés à certaines fonctions non-linéaires (une sous-classe des P-fonctions) : la propriété du "trou du pigeon" des points complémentaires. Cette propriété entraîne non seulement une nouvelle caractérisation des P-matrices, mais elle engendre une nouvelle famille d'algorithmes pour les problèmes de complémentarité : les algorithmes directs.

Deux classes de fonctions vérifient la propriété du "trou du pigeon" : les fonctions différentiables possédant une matrice jacobienne bornée positivement et les P-fonctions uniformes sur \mathbb{R}^n . Dans le cas des fonctions affines, les points complémentaires vérifient cette propriété si, et seulement si, tous les mineurs principaux de la matrice des coefficients sont positifs.

Nous illustrerons, au chapitre III, cette classe d'algorithmes par deux exemples : l'algorithme de Bard et celui de Murty [20]. La programmation de ces algorithmes nous a permis de constater que l'algorithme de Bard est le plus efficace en temps d'exécution.

Enfin, nous étudierons plus particulièrement un troisième algorithme de cette classe dont nous prouverons la convergence : l'algorithme de Kostreva [12]. Et, sur la base d'un test statistique : le test du signe, au chapitre IV, nous le comparerons aux deux précédents.

L E X I Q U E

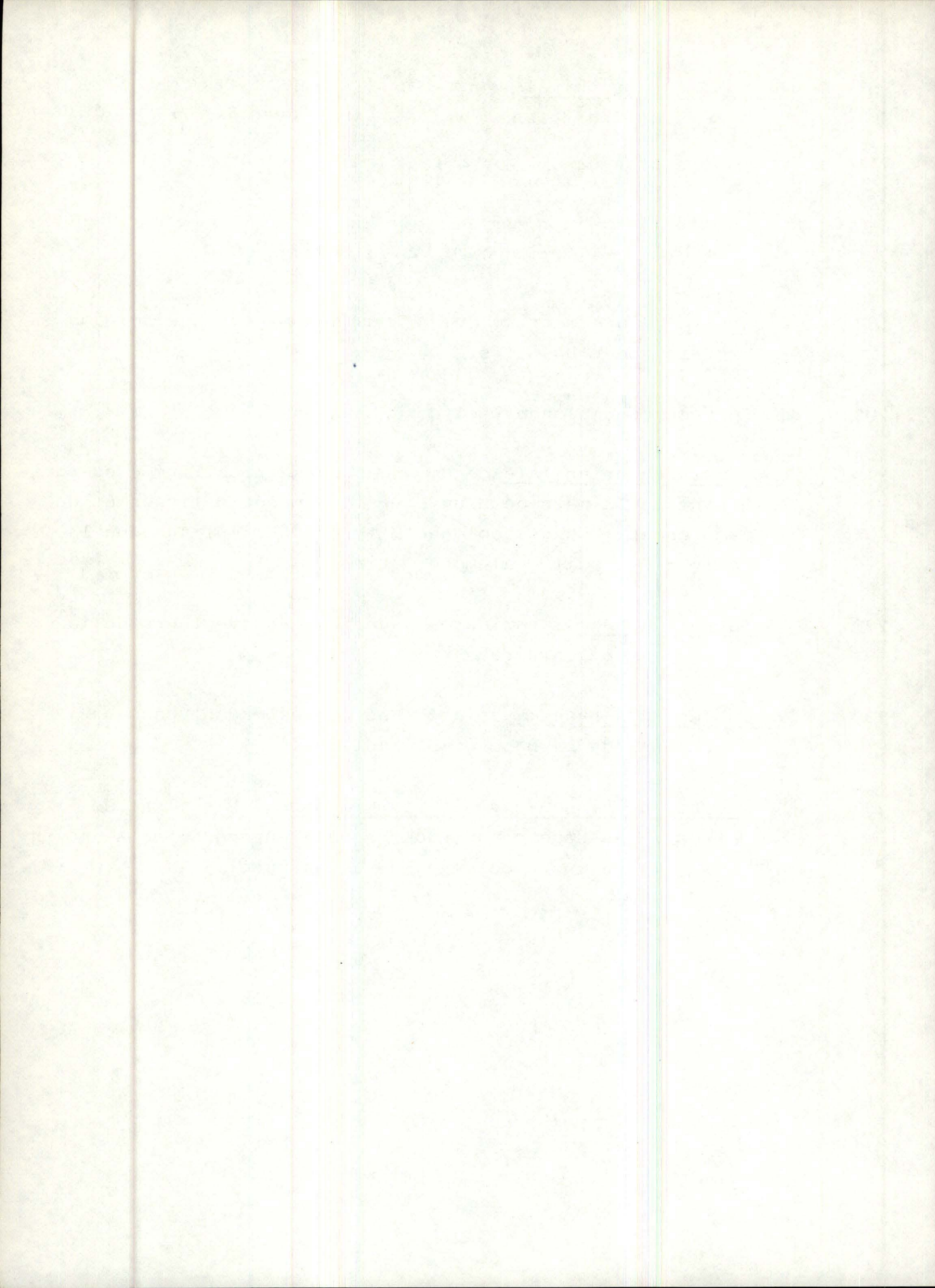
= = = = = = =

Outre les notations habituelles en théorie des matrices, nous utilisons dans le présent mémoire les notations et définitions suivantes :

- 1° Base complémentaire : un ensemble complémentaire de colonnes dont la matrice associée est inversible.
- 2° Cône complémentaire : le cône généré positivement par chaque ensemble complémentaire de colonnes.
- 3° $\mathcal{C}(M)$: la classe des cônes complémentaires associés à la matrice M .
- 4° Solution de base dégénérée : une solution de base dont la valeur d'une ou de plusieurs des variables est nulle.
- 5° Ensemble complémentaire de colonnes : un ensemble de n vecteurs colonnes $\{A_{.j} \mid j \in N\}$ tel que $A_{.j}$ est soit $I_{.j}$, soit $-M_{.j} \quad \forall j \in N$
- 6° LCP : un problème linéaire de complémentarité.
- 7° M : une matrice réelle $n \times n$.
- 8° La matrice carrée M est non-dégénérée : si tout ensemble complémentaire de colonnes est une base complémentaire.
- Remarquons que le déterminant d'une telle matrice est égal au déterminant de la sous-matrice principale de M correspondante. Donc M est une matrice \mathcal{L} -dégénérée si, et seulement si, tous les déterminants des sous-matrices principales sont non-nuls.

- 9° $\underline{N} \triangleq \{ 1, 2, \dots, n \}$
- 10° NLCP : un problème non-linéaire de complémentarité.
- 11° Orthant : le cône rectangulaire de l'espace \mathbb{R}^m généré par les vecteurs colonnes de la matrice unité. Ils sont au nombre de 2^n .
- 12° Orthant non-négatif : $\mathbb{R}_+^m \triangleq \{ x \mid x \in \mathbb{R}^m, x \geq 0 \}$
- 13° Paire de variables complémentaires : une paire de variables (w_i, z_i) vérifiant la condition de complémentarité :
- $$w_i z_i = 0 \quad \forall i \in N$$
- 14° PBJM : une classe de fonctions qui possèdent une matrice jacobienne $J_f(x)$ bornée positivement, c'est-à-dire $\exists \delta \in \mathbb{R}, 0 < \delta < 1$ tel que $\forall x \in \mathbb{R}^m$ tous les mineurs principaux de $J_f(x)$ soient compris entre δ et δ^{-1} .
- 15° P - fonction : une classe de fonctions qui vérifient la propriété :
- $$\exists i \quad (x_i - y_i) (f_i(x) - f_i(y)) > 0$$
- 16° P - fonction uniforme : une classe de fonctions qui vérifient la propriété :
- $$\max_{i \in N} (x_i - y_i) (f_i(x) - f_i(y)) \geq c \|x - y\|^2 \quad \forall c > 0$$
- 17° Pivotage : une suite d'opérations élémentaires qui remplacent un système par un autre équivalent (c'est-à-dire ayant même ensemble de solutions) dans lequel on a échangé les rôles d'une variable indépendante et d'une variable de base.

- 18° un Pivotage principal consiste à échanger les rôles de p variables de base w_{i_1}, \dots, w_{i_p} avec les p variables hors base z_{i_1}, \dots, z_{i_p}
- Attention, il n'est pas dit que successivement, chaque w_{i_j} est remplacé par le z_{i_j} correspondant.
- 19° P-matrice : une matrice qui a tous ses mineurs principaux strictement positifs.
- 20° q : un vecteur colonne réel de dimension n .
- 21° Sous-matrice principale de M , une sous-matrice formée en retirant de la matrice M un certain nombre de lignes et de colonnes. Chaque colonne M_j retirée implique que la ligne M_j soit également retirée.
- 22° Variables de base : variables associées aux vecteurs de la base complémentaire.
- 23° Variables hors-base : variables ne correspondant pas aux vecteurs de la base complémentaire.
- 24° Le vecteur q est non-dégénéré par rapport à M s'il ne se trouve pas dans chaque sous-espace engendré par au moins $(n - 1)$ vecteurs colonnes de l'ensemble
- $$\{I_{.1}, I_{.2}, \dots, I_{.m}, -M_{.1}, \dots, -M_{.m}\}$$



C H A P I T R E I

NOTIONS FONDAMENTALES
=====

1. POSITION DU PROBLEME

L'étude, par exemple, d'un programme linéaire ou d'un programme quadratique nous amène à considérer le problème suivant :

Etant donnés une matrice M appartenant à $\mathbb{R}^{m \times m}$ et un vecteur-colonne q appartenant à $\mathbb{R}^{m \times 1}$

Trouver, s'ils existent, des vecteurs w et z de l'espace \mathbb{R}^m qui satisfont, l'équation $w = q + Mz$

$$\left. \begin{array}{l} \text{et les conditions } w \geq 0, z \geq 0 \quad (i) \\ w^T z = 0 \quad (ii) \end{array} \right\} (0)$$

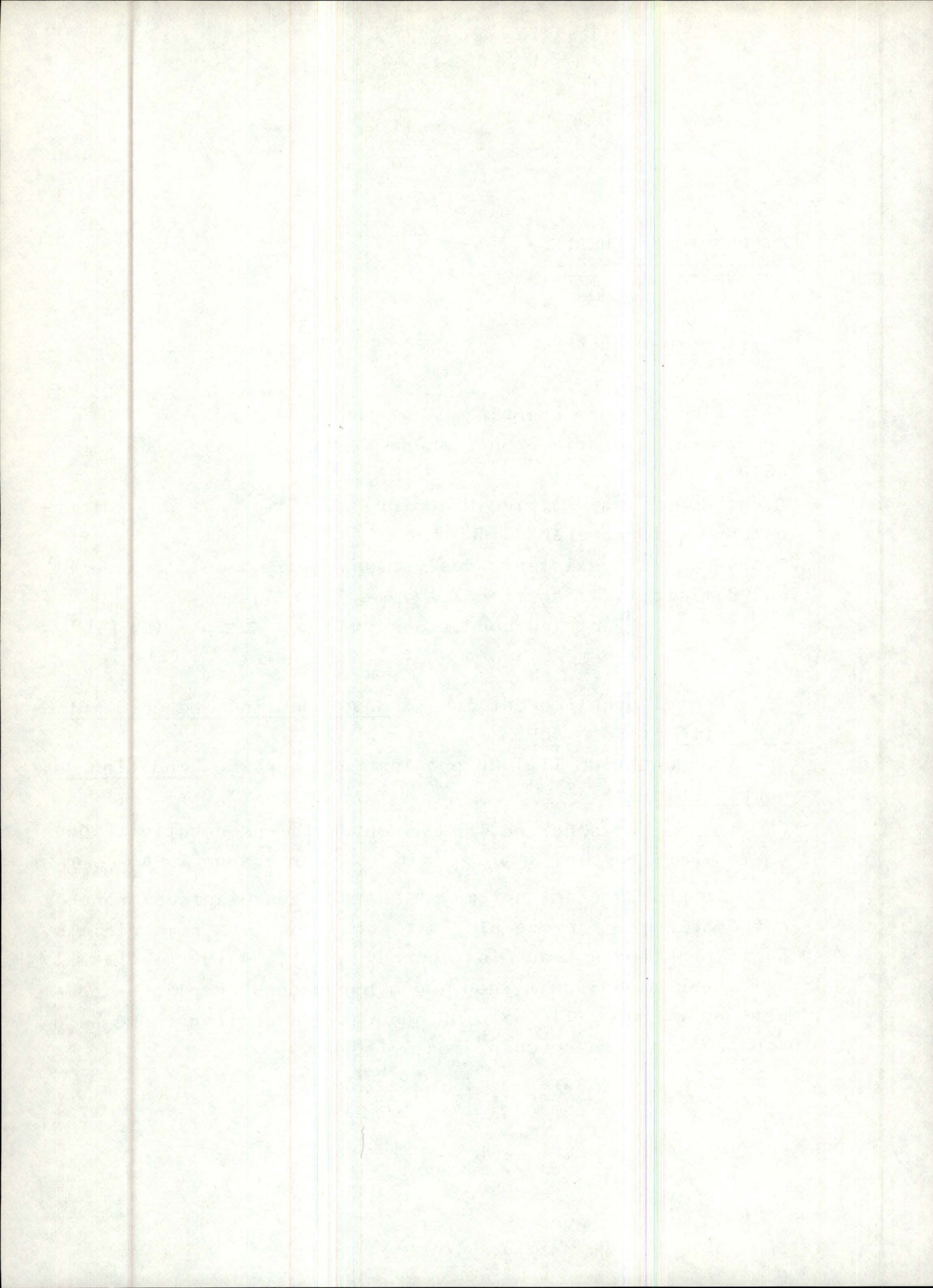
Un tel problème est appelé problème linéaire de complémentarité et noté LCP.

La condition (ii) du problème est appelée condition de complémentarité.

Elle peut s'écrire, en raison de la non-négativité des vecteurs w et z : $w_\lambda z_\lambda = 0$ pour chaque $\lambda = 1, \dots, m$

Ce problème est notamment abordé dans d'autres branches mathématiques, par exemple la théorie des jeux bimatriciels, où le problème a trouvé sa source. [3], [4], [5], [14], [23]

Dans ce qui suit, nous nous bornerons à expliquer comment un programme linéaire ou quadratique se transforme en un problème linéaire de complémentarité.



a) Considérons premièrement le programme linéaire sous la forme symétrique primale-duale [21].

Programme primal :

Minimiser la fonction $\bar{z} = c^T x$ pour x appartenant à \mathbb{R}^m et vérifiant les conditions :

$$\left. \begin{array}{l} Ax \geq b \\ x \geq 0 \end{array} \right\} \quad (1)$$

Programme dual :

Maximiser la fonction $z = y^T b$ pour y appartenant à \mathbb{R}^m et vérifiant les conditions :

$$\left. \begin{array}{l} yA \leq c \\ y \geq 0 \end{array} \right\} \quad (2)$$

Le théorème de dualité en programmation linéaire [7] nous permet d'affirmer que

$$\min \bar{z} = \max \underline{z}$$

lorsque les deux programmes possèdent des solutions réalisables optimales.

C'est pourquoi, comme pour toute solution réalisable du programme primal et pour toute solution réalisable du programme dual

$$z = y^T b \leq y^T A x \leq c^T x = \bar{z}$$

on cherche une solution telle que

$$y^* b = c^T x^* \quad (3)$$

Cependant, on peut transformer les contraintes d'inégalité des programmes primal et dual en un système équivalent d'équations composées de variables non-négatives par adjonction de variables d'écart.

Les systèmes (1) et (2) sont donc équivalents au système

$$\left. \begin{array}{l} A x - r = b \quad r \geq 0, \quad x \geq 0 \\ A^T y + \mu = c \quad \mu \geq 0, \quad y \geq 0 \end{array} \right\} \quad (4)$$

Le problème de programmation linéaire revient alors à trouver les vecteurs μ, r, x, y tels que

$$\begin{pmatrix} \mu \\ r \end{pmatrix} = \begin{pmatrix} c \\ -b \end{pmatrix} + \begin{pmatrix} 0 & -A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

$$\mu \geq 0, \quad r \geq 0$$

$$x \geq 0, \quad y \geq 0$$

et, par (3)

$$r \mu + y r = 0$$

Si l'on pose

$$w = \begin{pmatrix} \mu \\ r \end{pmatrix}, \quad q = \begin{pmatrix} c \\ -b \end{pmatrix}$$

$$M = \begin{pmatrix} 0 & -A^T \\ A & 0 \end{pmatrix}, \quad z = \begin{pmatrix} x \\ y \end{pmatrix}$$

On obtient alors le problème (o).

b) Le problème de programmation quadratique s'énonce de la manière suivante:

Minimiser la fonction $\bar{z} = c^T x + \frac{1}{2} x^T D x$ pour x appartenant à \mathbb{R}^m et vérifiant les conditions :

$$\left. \begin{array}{l} Ax \geq b \\ x \geq 0 \end{array} \right\} \quad (5)$$

Nous pouvons supposer que la matrice D est symétrique.

En effet, si ce n'est le cas, on remplace D par $(D+D^T)/2$, ce qui laisse la fonction \bar{z} inchangée.

Lorsque la fonction à minimiser est une fonction convexe en x (c'est-à-dire si la matrice D est semi-définie positive), le problème (5) est un problème de programmation quadratique convexe.

Pour ce problème, définissons μ et ν par

$$\left. \begin{array}{l} \mu = Dx - A^T y + c \\ \nu = Ax - b \end{array} \right\} \quad (6)$$

Dans ces conditions, le vecteur x^0 minimise la fonction \bar{z} s'il existe un vecteur y^0 et des vecteurs μ^0, ν^0 définis par (6) pour $x = x^0$ satisfaisant

$$\begin{aligned} x^0 \geq 0, \quad \mu^0 \geq 0, \quad y^0 \geq 0, \quad \nu^0 \geq 0 \\ x^0 \mu^0 = 0, \quad y^0 \nu^0 = 0 \end{aligned}$$

Ces conditions nécessaires, caractérisant le minimum du problème de programmation quadratique (1), sont des conséquences directes du théorème de Kuhn et Tucker [13].

Ce sont les conditions de Kuhn-Tucker qui, on le sait, sont suffisantes dans le cas de programmation quadratique convexe.

Le problème de programmation quadratique revient donc à résoudre le système :

$$\begin{aligned} \mu &= Dx - A^T y + c & x \geq 0, y \geq 0 \\ \nu &= Ax - b & \mu \geq 0, \nu \geq 0 \end{aligned}$$

$$x\mu + y\nu = 0$$

En posant $w = \begin{pmatrix} \mu \\ \nu \end{pmatrix}$, $q = \begin{pmatrix} c \\ -b \end{pmatrix}$, $M = \begin{pmatrix} D & -A^T \\ A & 0 \end{pmatrix}$, $z = \begin{pmatrix} x \\ y \end{pmatrix}$

On obtient le problème (0).

Définissons, en illustrant d'un exemple, les concepts importants du problème.

Soient $m = 2$

$$M = \begin{pmatrix} -2 & -1 \\ -4 & 0 \end{pmatrix}$$

$$q = \begin{pmatrix} 2 \\ 3 \end{pmatrix}$$

Dans ce cas, le problème s'écrit :

$$w_1 + 2z_1 + z_2 = 2$$

$$w_2 + 4z_1 = 3$$

(1.1)

$$w_1, w_2, z_1, z_2 \geq 0 \quad (i)$$

et $w_1 z_1 = w_2 z_2 = 0 \quad (ii)$

Exprimé, sous forme matricielle, le problème (1.1) devient

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} + \begin{pmatrix} 2 & 1 \\ 4 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \end{pmatrix} \quad (1.2)$$

$$\left. \begin{aligned} w_1, w_2, z_1, z_2 &\geq 0 \\ w_1 z_1 &= w_2 z_2 = 0 \end{aligned} \right\} \quad (1.3)$$

Pour chaque solution satisfaisant (1.3), au moins une des variables de chaque paire (w_λ, z_λ) , (w_2, z_2) est nulle. Ces paires (w_λ, z_λ) sont appelées paires de variables complémentaires dont la définition exacte est : une paire de variables (w_λ, z_λ) vérifiant la condition de complémentarité : $w_\lambda z_\lambda = 0 \quad \forall \lambda \in N$ est une paire de variables complémentaires.

Une première approche pour résoudre ce problème est de prendre une variable dans chaque paire complémentaire et de fixer sa valeur à zéro dans (1.2).

Nous obtenons alors un ensemble complémentaire de colonnes C, c'est-à-dire un ensemble de n vecteurs colonnes

$$\{A_j \mid j \in N\} \text{ tel que } A_j \text{ est soit } I_j, \text{ soit } -M_j \quad \forall j \in N$$

Si la matrice formée à partir de cet ensemble est inversible, nous parlons de base complémentaire BC.

Exemple : Pour la matrice M considérée ci-dessus, nous pourrions avoir :

$$C = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\}$$

$$BC = C = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$$

$$BC = C = \left\{ \begin{pmatrix} 2 \\ 4 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$$

Les variables ne correspondant pas à la base complémentaire sont annulées et appelées : variables hors-base, par opposition aux autres, appelées variables en-base.

Après élimination des variables nulles dans (1.2) et si le système possède une solution dont les variables en-base sont non négatives, le problème linéaire de complémentarité sera résolu.

Choisissons par exemple, w_1 et w_2 comme variables à annuler dans (1.2). Le système (1.1) dont les variables en-base sont z_1 et z_2 devient alors :

$$\left. \begin{aligned} z_1 \begin{pmatrix} 2 \\ 4 \end{pmatrix} + z_2 \begin{pmatrix} 1 \\ 0 \end{pmatrix} &= \begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = q \\ z_1 \geq 0, z_2 \geq 0 \end{aligned} \right\} \quad (1.4)$$

On constate que l'équation (1.4) possède une solution si le vecteur q peut s'exprimer comme combinaison linéaire non négative des vecteurs $\begin{pmatrix} 2, 4 \end{pmatrix}^T$ et $\begin{pmatrix} 1, 0 \end{pmatrix}^T$.

D'autre part, l'ensemble de toutes les combinaisons linéaires non négatives de $\begin{pmatrix} 2, 4 \end{pmatrix}^T$ et de $\begin{pmatrix} 1, 0 \end{pmatrix}^T$ est un cône dans l'espace q_1, q_2 (fig. 1).

Dès lors, le LCP possèdera une solution dont les variables en-base sont z_1, z_2 seulement si le vecteur q égal à $\begin{pmatrix} 2, 3 \end{pmatrix}^T$ se trouve dans ce cône.

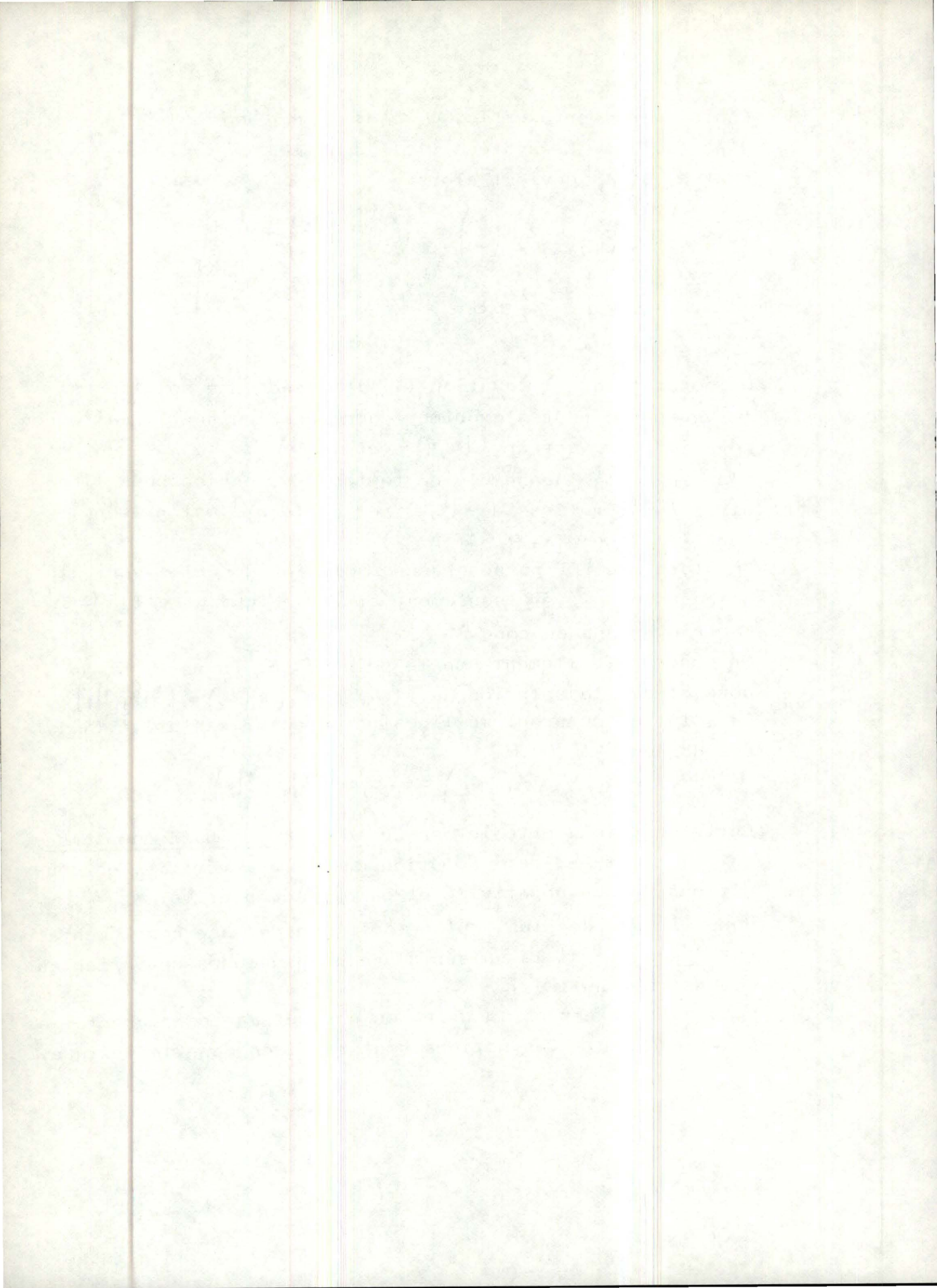
On observe facilement que le point $\begin{pmatrix} 2, 3 \end{pmatrix}^T$ appartient au cône et que la solution de (1.4) est $(z_1, z_2) = (3/4, 1/2)$. La solution du problème linéaire de complémentarité (1.1) est donc :

$$(w_1, w_2, z_1, z_2) = (0, 0, 3/4, 1/2)$$

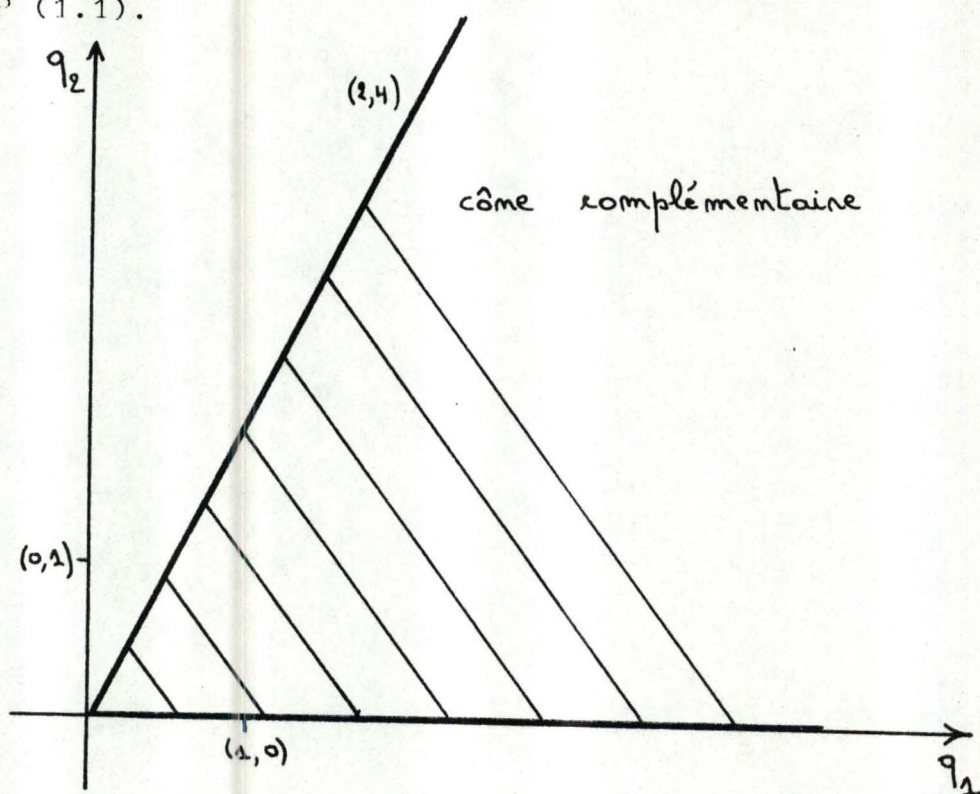
Cette solution porte le nom de solution de base complémentaire, c'est-à-dire: solution du LCP qui vérifie les conditions de non-négativité et de complémentarité.

Nous dirons, de plus, qu'une solution de base complémentaire est dégénérée si la valeur d'une ou plusieurs de variables de base est nulle.

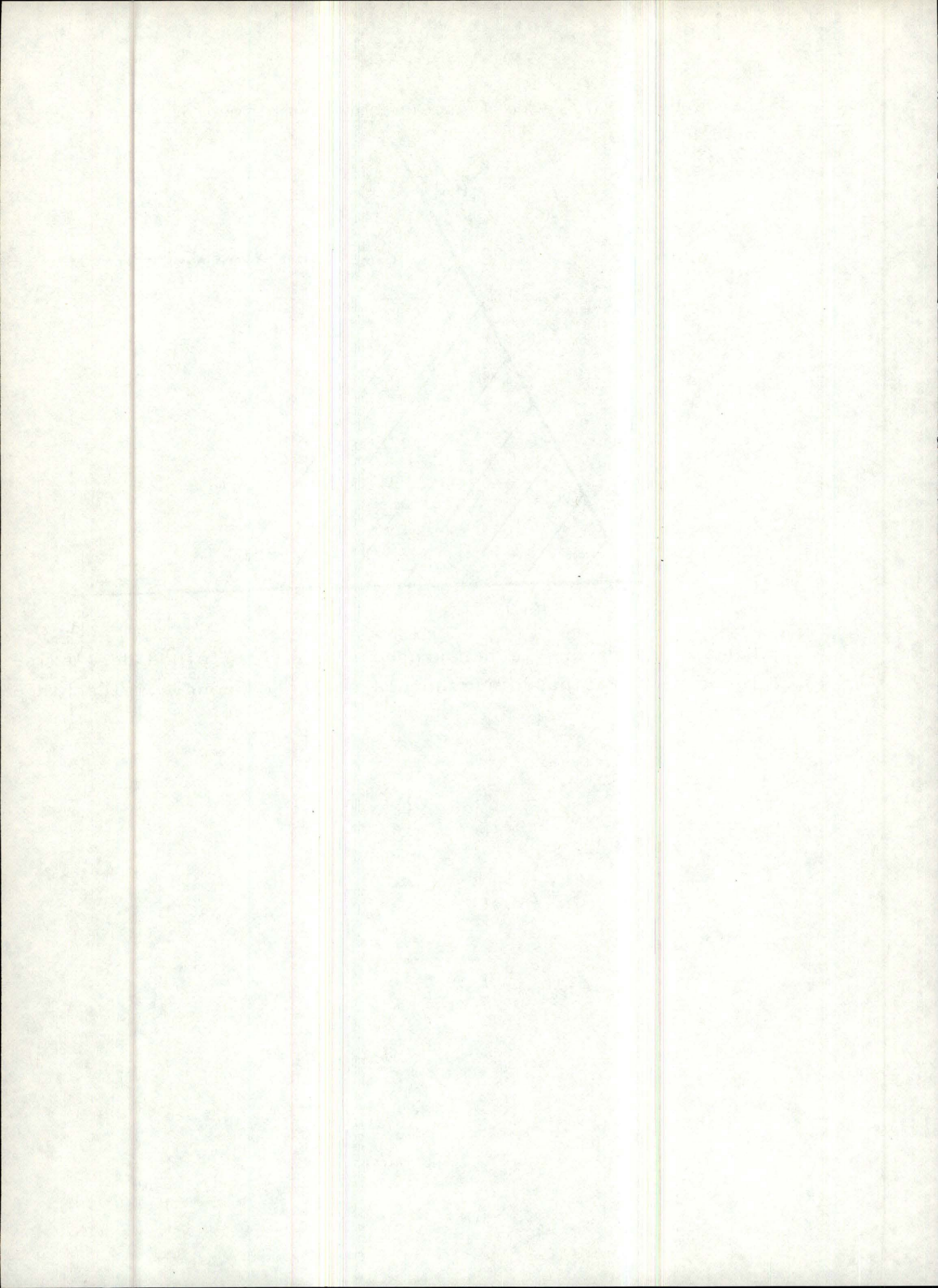
Dans notre exemple, la solution est de base complémentaire non-dégénérée. Ces notions sont reprises dans le lexique.



Le cône de la figure 1 est un cône complémentaire associé au LCP (1.1).



Nous consacrerons le paragraphe suivant à l'étude de l'existence de solutions utilisant la notion de cône complémentaire.



2.

INTERPRETATION GEOMETRIQUE DES CONES COMPLEMENTAIRES

Le problème linéaire de complémentarité s'écrit sous forme matricielle de la manière suivante :

$$(\mathbb{I} ; -M) \begin{pmatrix} w \\ z \end{pmatrix} = (q)$$

$$w^T z = 0$$

$$w \geq 0, z \geq 0$$

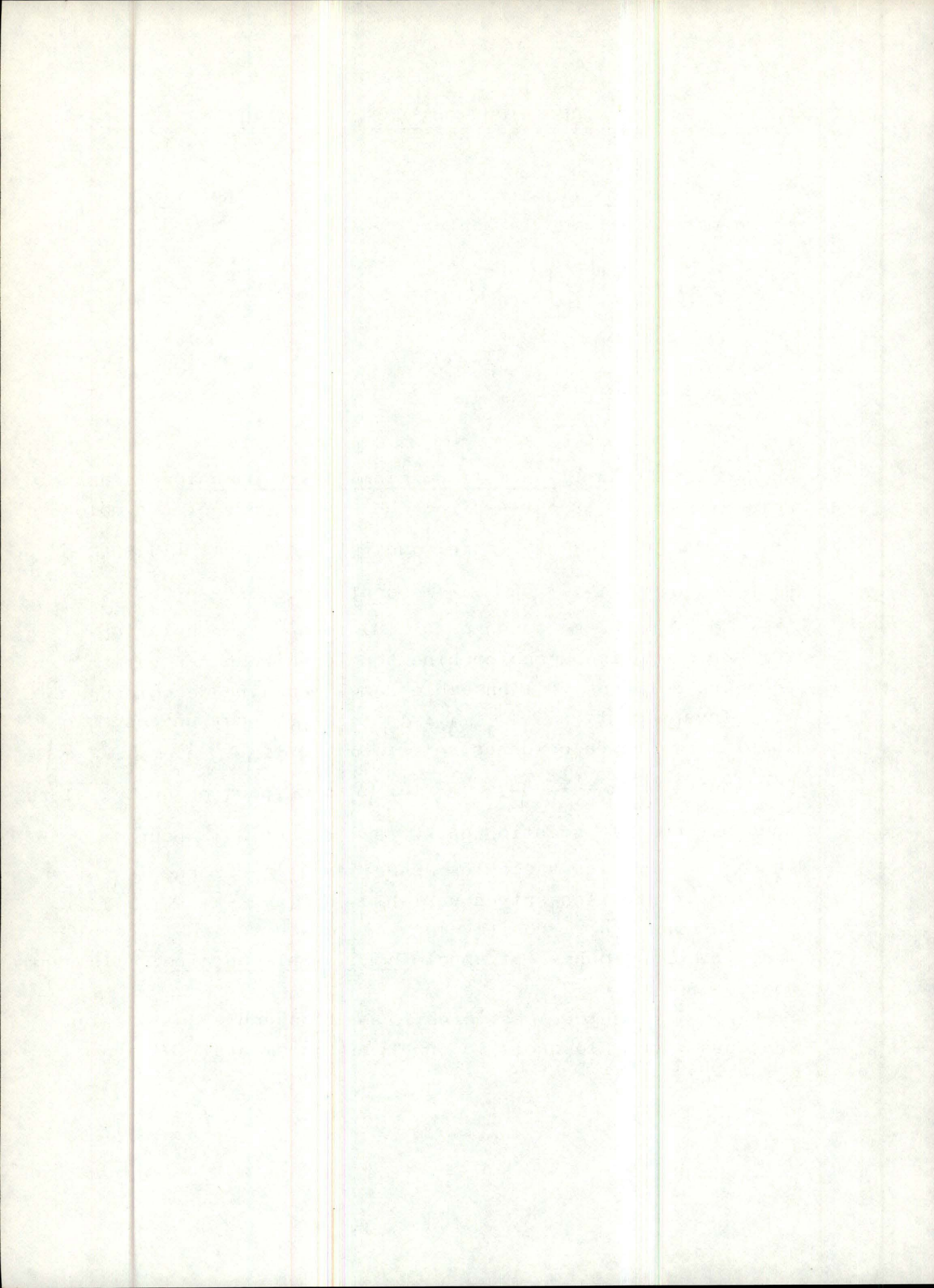
Considérons l'ensemble complémentaire de colonnes associé à la matrice M . C'est-à-dire un ensemble de vecteurs colonnes $\{A_j \mid j=1, \dots, m\}$ tel que A_j est soit \mathbb{I}_j soit $-M_j$ pour chaque $j=1, \dots, m$. Ils sont au nombre de 2^n .

Notre problème fondamental (0) possède une solution si le vecteur q s'exprime comme combinaison linéaire non-négative des vecteurs colonnes de l'ensemble complémentaire de colonnes.

Inversément, si $\{A_j \mid j=1, \dots, m\}$ est un ensemble complémentaire de colonnes, et si $q \in \text{pos} \{A_j \mid j=1, \dots, m\}$ c'est-à-dire $q = \sum_{j=1}^m \beta_j A_j$ où $\beta_j \geq 0$ pour chaque j , alors nous obtenons la solution du LCP en égalant à β_j pour $j=1, \dots, m$ les variables associées à la colonne A_j et en annulant toutes les autres variables de (w, z)

Le cône généré positivement par chaque ensemble complémentaire de colonnes est appelé cône complémentaire. Ils sont donc au nombre de 2^n .

L'union de tous ces cônes forme l'ensemble de tous les vecteurs q pour lesquels le problème fondamental (0) a une solution.



C'est pourquoi tous les résultats relatifs au nombre de solutions distinctes d'un problème de complémentarité peuvent être interprétés et développés en termes de propriétés de cônes complémentaires et vice-versa.

On notera, dès lors, $\mathcal{C}(M)$ la classe des cônes complémentaires associés à la matrice M .

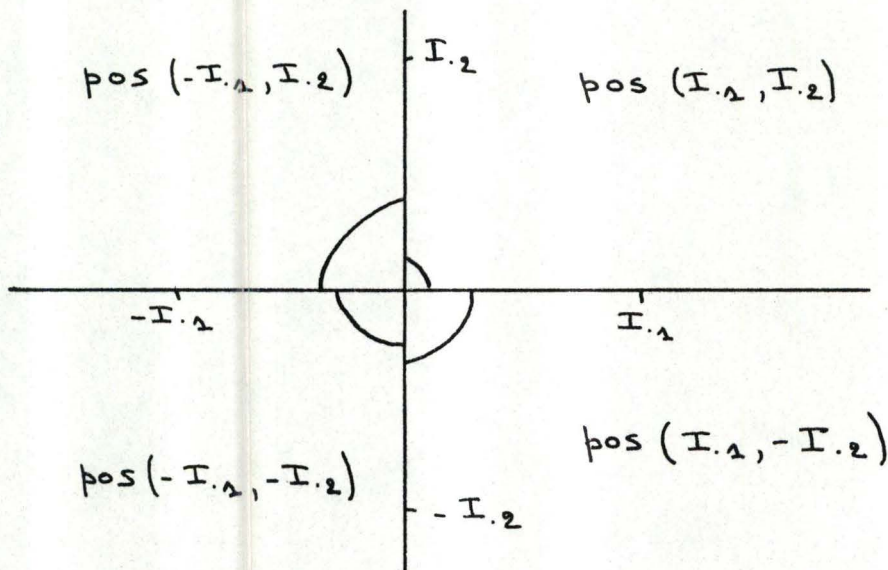
Nous illustrerons cette notion de cônes complémentaires par trois exemples :

Exemple 1

soient $m=2$ et $M = \mathbb{1}_{\mathbb{R}^2}$

Dans ce cas, la classe $\mathcal{C}(I)$ est la classe des orthants de \mathbb{R}^2 (voir lexique).

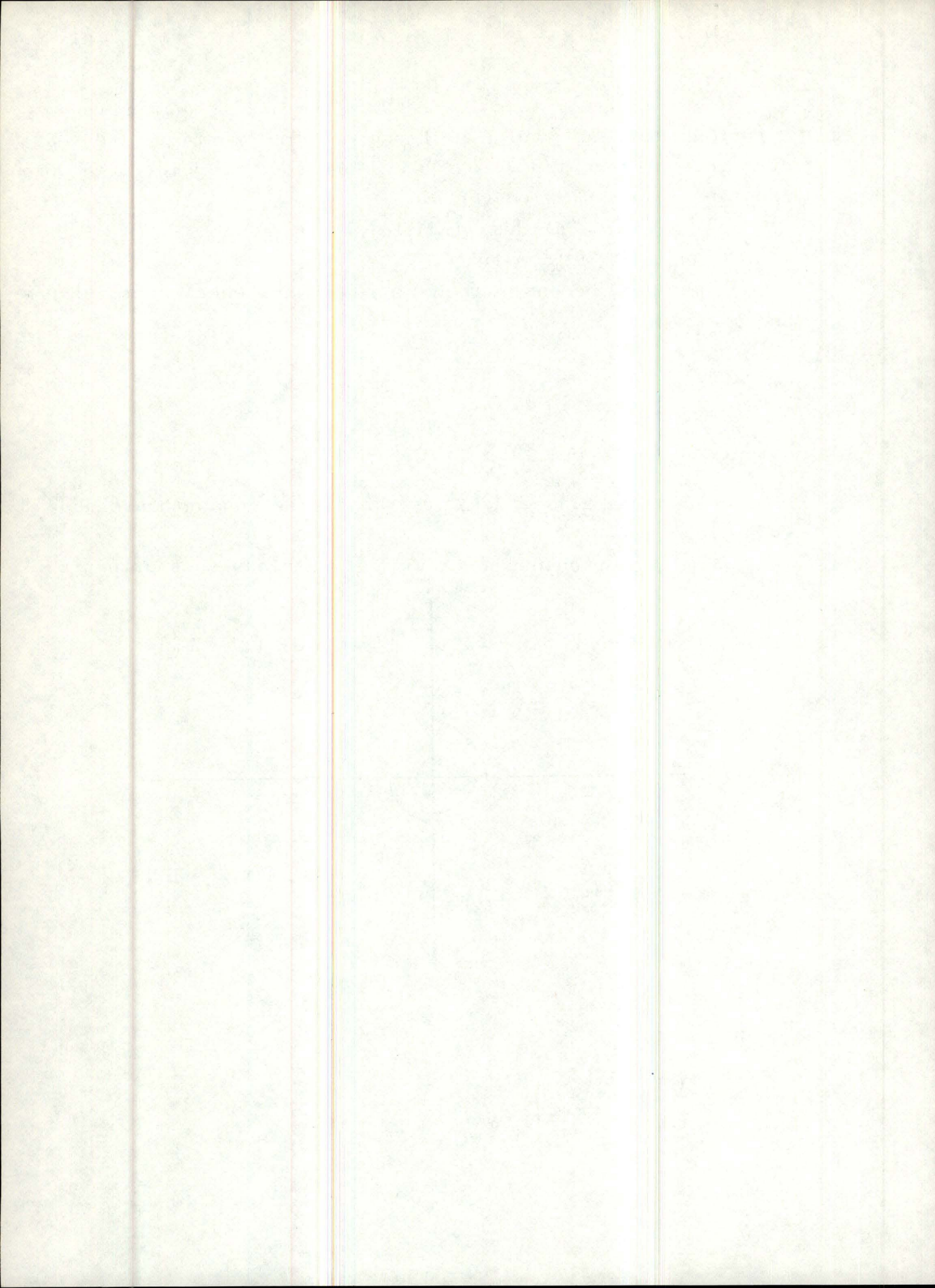
En général, pour chaque m , $\mathcal{C}(I)$ est la classe des orthants de \mathbb{R}^m .

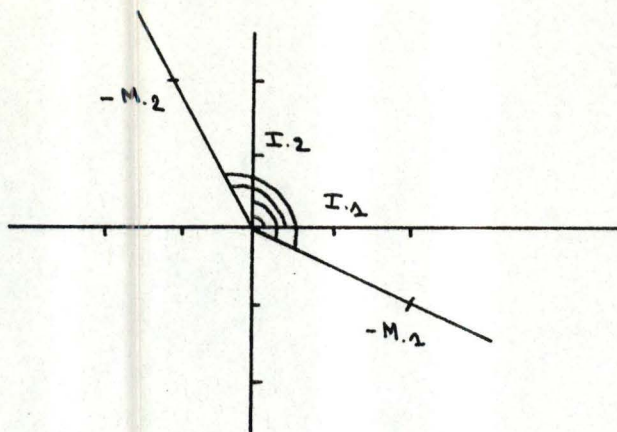


Exemple 2

Soient $m=2$

$$\text{et } M = \begin{pmatrix} -2 & 1 \\ 1 & -2 \end{pmatrix}$$

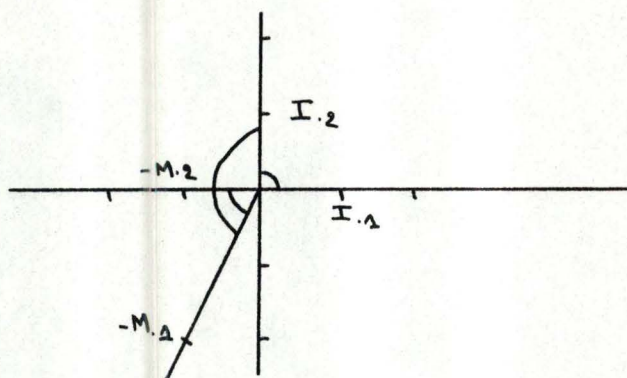




Exemple 3

Soient $m=2$ et $M = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$

On observe que $\{I.1, -M.2\}$ est un ensemble linéairement dépendant et que l'intérieur du cône $\text{pos}(I.2, -M.2)$ est vide.



Résoudre un LCP revient donc à trouver le cône complémentaire dans $\mathcal{C}(M)$ qui contient le point q .

Chaque solution d'un LCP correspond donc à la représentation de q comme combinaison linéaire non-négative de $(A.1, \dots, A.m)$.

Dès lors, un LCP possède une solution si, et seulement si, q se trouve dans l'union des 2^n cônes complémentaires. La solution est unique si q se trouve exactement dans un seul cône complémentaire.

Cette approche, via les cônes complémentaires, a été fort utile dans l'étude des problèmes d'existence et d'unicité des solutions.

C'est pourquoi nous allons, dans le paragraphe suivant, énoncer quelques théorèmes importants d'existence et d'unicité des solutions.

3.

THEOREMES D'EXISTENCE ET D'UNICITE [8], [9]

Lemke [16] a montré que le nombre de solutions de base réalisables complémentaires est fini lorsque le vecteur q est non-dégénéré par rapport à la matrice M (définition : voir lexique).

Ici, nous nous limiterons à énoncer les conditions nécessaires et suffisantes sous lesquelles le nombre de solutions du problème fondamental (0) est fini quel que soit q , élément de \mathbb{R}^m .

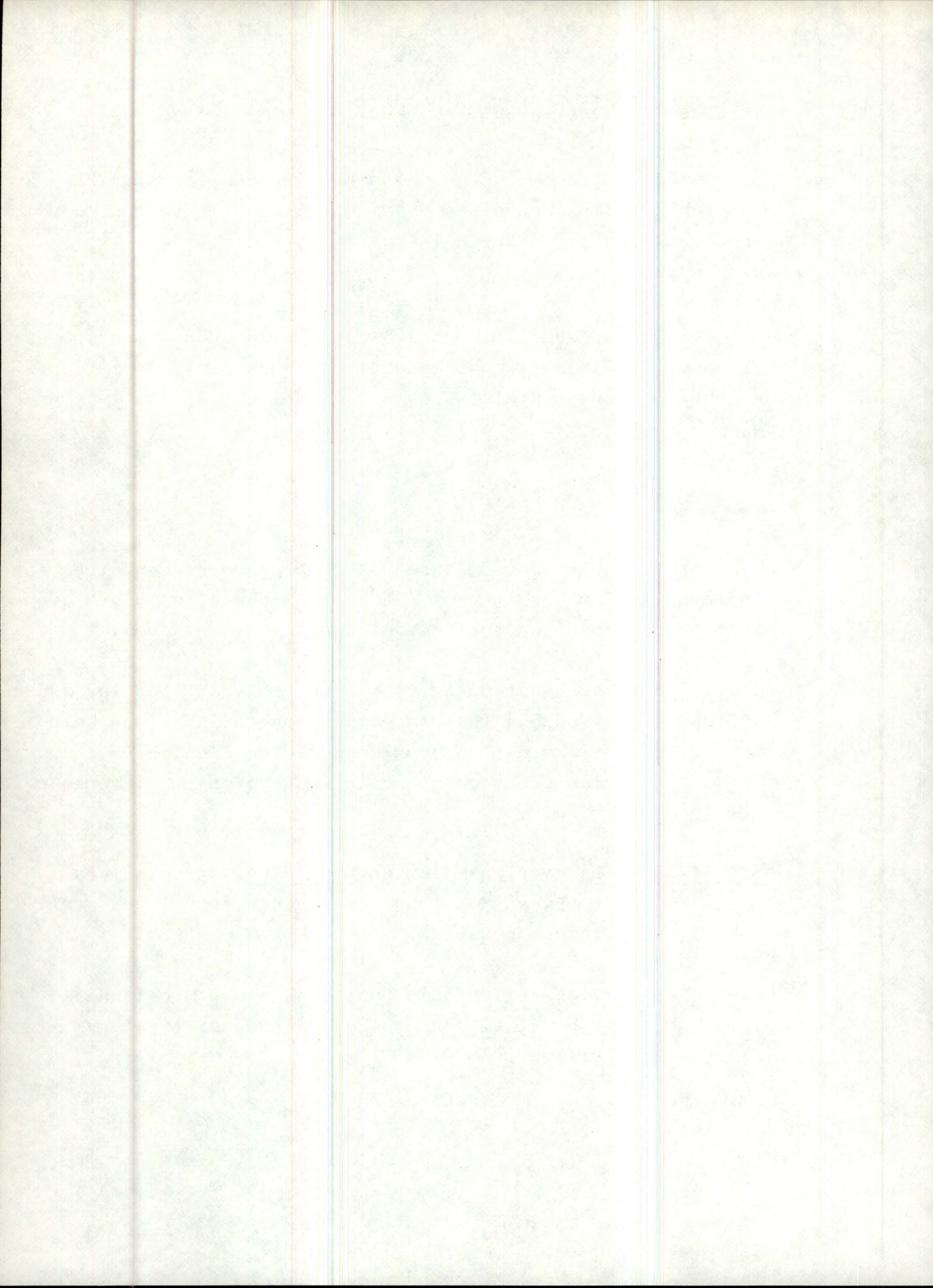
Théorème 1

Le nombre de solutions de base réalisables complémentaires est fini pour tout q appartenant à \mathbb{R}^m si, et seulement si, M est une matrice non-dégénérée (définition : voir lexique).

Corollaire : si M est dégénérée, l'ensemble de tous les q pour lesquels (0) possède un nombre infini de solutions est un sous-ensemble de l'union de tous les cônes complémentaires qui ont un intérieur vide.

Définition : La matrice M est un P-Matrice, et on notera $M \in P$, si M a tous ses mineurs principaux strictement positifs.

Exemple : Si M est définie positive, alors M est une P-Matrice. On a l'équivalence si M est symétrique.



Théorème 2

Le système (0) possède une solution unique pour chaque $q \in \mathbb{R}^m$ si, et seulement si, M est une P-Matrice.

Corollaire : L'ensemble des cônes complémentaires forme une partition de \mathbb{R}^m si, et seulement si, M est une P-Matrice.

Théorème 3

Si M est non-dégénérée, alors le nombre de solutions de base réalisables complémentaires a la même parité pour tout $q \in \mathbb{R}^m$ qui est non-dégénéré par rapport à M .

Pour les démonstrations de ces théorèmes : voir [19].

4.

QUELQUES ALGORITHMES DISPONIBLES [1] , [15] , [20]Algorithme de Lemke

Lemke est un des premiers à avoir mis au point une méthode pour résoudre de manière systématique le problème de complémentarité (0).

Pour démarrer son algorithme, on adjoint au problème un vecteur colonne.

Souvent on prend $\alpha = (\alpha_1 \ \alpha_2 \ \dots \ \alpha_n)^T$

Le choix de ce vecteur est cependant essentiel pour la terminaison de l'algorithme [15].

L'algorithme de Lemke est intéressant à analyser comme technique de résolution. C'est pourquoi nous y consacrerons l'essentiel de cette partie de notre étude, avant d'aborder des algorithmes d'un type différent : les algorithmes directs.

Données : soit le problème de complémentarité à résoudre :

$$w = q + Mz$$

$$w^T z = 0$$

$$w \geq 0, z \geq 0$$

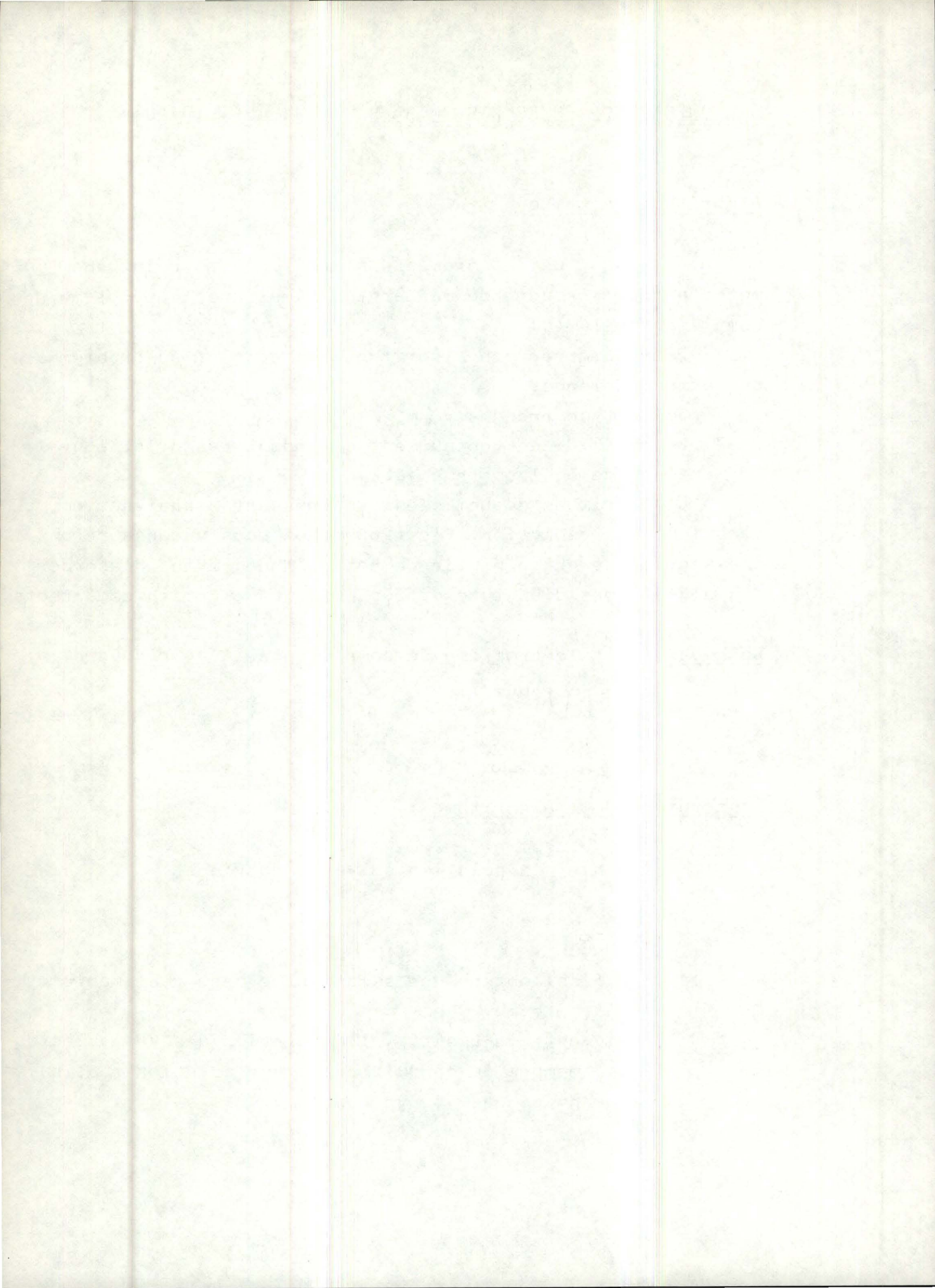
Description de l'algorithme :

Pas 1 - Si $q \geq 0$, la solution suivante convient :

$$\begin{cases} w = q \\ z = 0 \end{cases}$$

On a ainsi obtenu une solution de base réalisable et on s'arrête.

Si q est non-positif, l'annulation des variables du second membre ne conduit pas à une solution réalisable. Dans ce cas, on va au pas 2.



Pas 2 - On adjoint au problème une variable artificielle z_0

On adjoint également un vecteur $\alpha > 0$ (par exemple $\alpha = (\alpha_1 \alpha_2 \dots \alpha_n)^T$) pour obtenir le problème augmenté :

$$\begin{aligned} w &= q + Mz + \alpha z_0 \\ &= q + A \hat{z} \end{aligned}$$

avec $\hat{z} = (z, z_0)^T$

$$A = (M, \alpha)$$

Pas 3 - Partant de la solution de base correspondante, on fait croître z_0 jusqu'à la valeur $\max_{q_i < 0} \left(\frac{-q_i}{\alpha_i} \right)$

Supposons que ce minimum ait lieu pour $i = k$; on effectue alors une opération de pivotage. On échange les rôles de z_0 et de w_k (voir la description précise ci-dessous). La variable z_0 va donc entrer en base et la variable w_k va en sortir.

Dès lors, dans la nouvelle forme de base,

- les variables de base, toutes positives, sont :

$$w_1, \dots, w_{k-1}, z_0, w_{k+1}, \dots, w_m$$

- les variables hors base sont :

$$z_1, \dots, z_m, w_k$$

et donc $w^1 = (w_1, \dots, z_0, \dots, w_m)^T = q^1$

$$z^1 = (z_1, \dots, z_m, w_k)^T = 0$$

est la nouvelle solution de base réalisable.

Notons que généralement w^i désignera l'ensemble des variables de base à l'itération i ,

z^i désignera l'ensemble des variables hors base à l'itération i ,

A^i désignera la matrice des coefficients du système d'équations à l'itération i .

Pas 4 - A partir de cette nouvelle solution de base, on augmente la valeur de la variable hors base z_k , complémentaire de la variable w_k distinguée en pas 3.

Pas 5 - Si la valeur de la variable z_k peut être augmentée indéfiniment sans qu'aucune des variables de base ne s'annule (c'est le cas si le vecteur colonne de la matrice A^2 correspondant à la variable z_k est positive ou nulle), l'algorithme s'arrête dans l'indécision.

Pas 6 - Si l'augmentation de la valeur de la variable z_k est bloquée par l'annulation d'une des variables de base soit w_λ^1 , deux possibilités se présentent :

- ou bien $w_\lambda^1 = z_0$, et le programme se termine avec, comme solution de base réalisable, la solution

$$\begin{cases} w^2 = q^2 \\ z^2 = 0 \end{cases}$$

- où bien $w_\lambda^1 \neq z_0$ et on recommence au pas 3, en faisant croître z_λ^1 . L'indice λ est déterminé de la manière suivante : c'est l'indice qui conduit à la première annulation d'une variable de base.

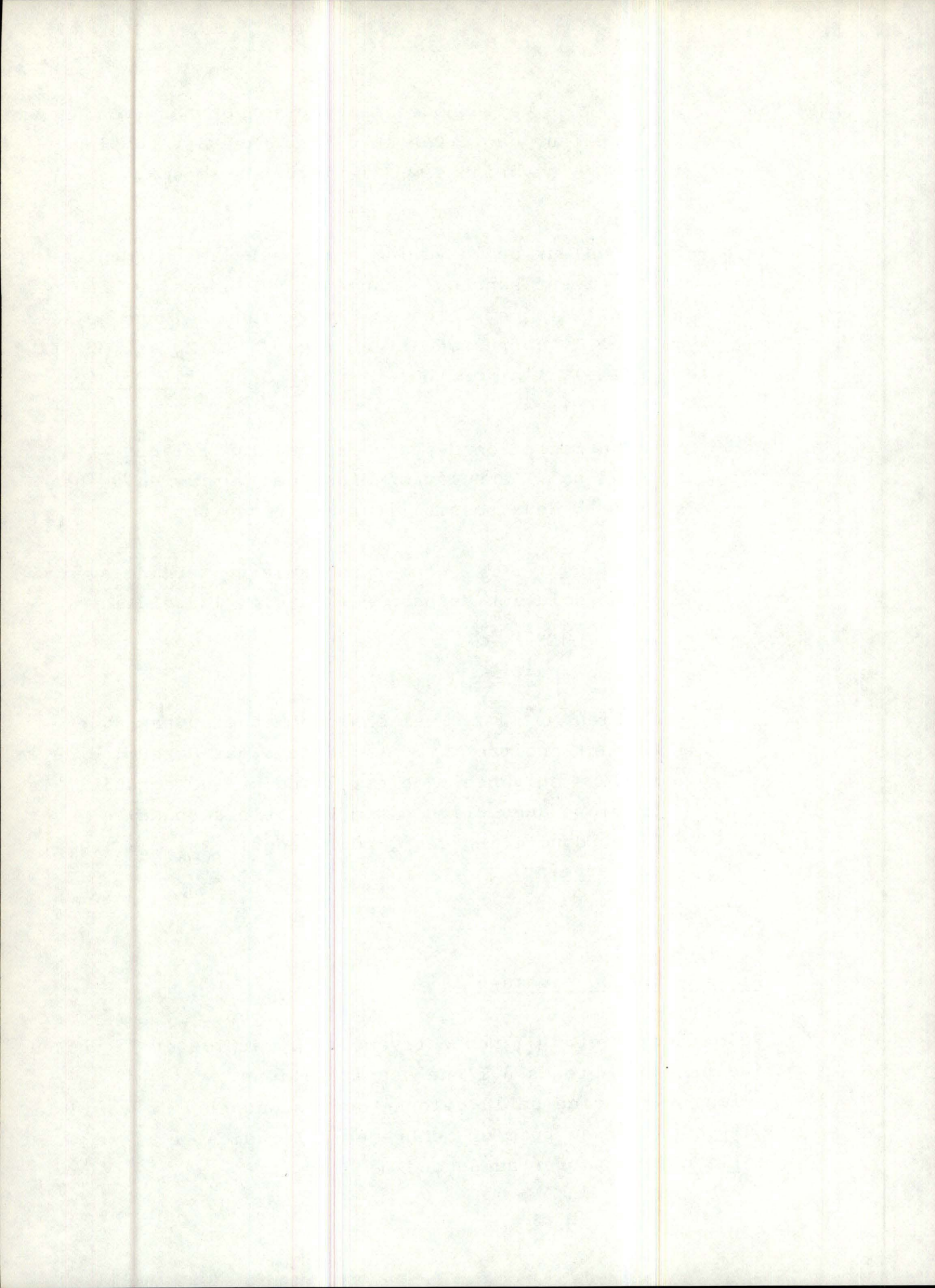
C'est donc l'indice i , pour lequel est atteint.

$$\min_{i \mid a_{i\lambda} < 0} \frac{-q_i}{a_{i\lambda}}$$

Description du pivotage :

Soient (a_{ij}) et (q_i) respectivement la matrice et le vecteur indépendant obtenus à l'une des itérations.

Soient λ l'indice de la colonne correspondant à la variable qui va passer au premier membre et k l'indice de ligne de celle qui va passer au second membre.



Le pivotage va donc s'effectuer sur l'élément a_{rs} du tableau

Le pivotage consiste à effectuer les opérations suivantes :

- le pivot est inversé $a_{rs} \rightarrow 1/a_{rs}$
- les éléments de la ligne du pivot (autres que celui-ci) sont divisés par l'opposé du pivot $a_{rj} \rightarrow -a_{rj}/a_{rs}$
- les éléments de la colonne du pivot (autres que celui-ci) sont divisés par le pivot $a_{is} \rightarrow a_{is}/a_{rs}$
- les autres éléments se transforment selon la formule $a_{ij} \rightarrow a_{ij} - (a_{rj} \cdot a_{is})/a_{rs}$

Exemples :

Les deux exemples présentés ci-dessous ont été choisis parce que le premier possède une terminaison normale et que le second s'arrête dans l'indécision. Ils ont été traités par le programme "Lemke" qui a été réalisé au cours de l'année académique 77-78 par un groupe d'étudiants composé de Bouillon André-Marie, Malherbe Anne-Marie, et Maquet Anne.

Exemple_1

Le tableau initial est :

	Z01	Z02	Z03	Z04		Q
W01	3.00	-2.00	-2.00	2.00	0.00	-1.00
W02	0.00	2.00	1.00	-5.00	0.00	6.00
W03	2.00	-1.00	0.00	0.00	0.00	-2.00
W04	-2.00	5.00	0.00	0.00	0.00	4.00

Le tableau augmenté est :

	Z01	Z02	Z03	Z04	Z.00	Q
W01	3.00	-2.00	-2.00	2.00	1.00	-1.00
W02	0.00	2.00	1.00	-5.00	1.00	6.00
W03	2.00	-1.00	0.00	0.00	1.00	-2.00
W04	-2.00	5.00	0.00	0.00	1.00	4.00

La colonne du pivot pour l'itération 1 est 5 (sa recherche est immédiate). La ligne du pivot pour l'itération 1 est 3.

Le tableau de l'itération 1 est :

	Z01	Z02	Z03	Z04	W03	Q
W01	1.00	-1.00	-2.00	2.00	1.00	1.00
W02	-2.00	3.00	1.00	-5.00	1.00	8.00
Z00	-2.00	1.00	0.00	0.00	1.00	2.00
W04	-4.00	6.00	0.00	0.00	1.00	6.00

La colonne du pivot pour l'itération 2 est 3.

La ligne du pivot pour l'itération 2 est 1.

Le tableau de l'itération 2 est :

	Z01	Z02	W01	Z04	W03	Q
Z03	0.50	-0.50	-0.50	1.00	0.50	0.50
W02	-1.50	2.50	-0.50	-4.00	1.50	8.50
Z00	-2.00	1.00	0.00	0.00	1.00	2.00
W04	-4.00	6.00	0.00	0.00	1.00	6.00

La colonne du pivot pour l'itération 3 est 1.

La ligne du pivot pour l'itération 3 est 3.

Le tableau de l'itération 3 est :

	Z00	Z02	W01	Z04	W03	Q
Z03	-0.25	-0.25	-0.50	1.00	0.75	1.00
W02	0.75	1.75	-0.50	-4.00	0.75	7.00
Z01	-0.50	0.50	0.00	0.00	0.50	1.00
W04	2.00	4.00	0.00	0.00	-1.00	2.00

La variable Z00 est repassée au second membre, la terminaison est normale. Ce tableau est donc le tableau final de l'algorithme.

Les solutions sont :

W	Z
0.00	1.00
7.00	0.00
0.00	1.00
2.00	0.00

Exemple 2

Le tableau initial est :

	Z01	Z02	Z03	Z04	Z05	Z06	Z07	Z08	Z09	Z10	Z11	Z12	Z13	e	
W01	1.00	-2.00	4.50	-6.75	-4.50	-2.50	-3.50	-3.75	-4.50	3.25	-3.50	-3.50	-1.25	0.00	2.50
W02	3.50	-6.50	-5.00	10.50	-6.50	-9.25	4.50	6.25	0.00	5.00	2.25	2.00	-5.75	0.00	-3.75
W03	4.50	-3.75	-5.00	12.50	-4.50	-3.00	5.25	5.00	0.25	0.00	0.00	1.00	5.00	0.00	12.00
W04	-12.50	-4.50	0.00	0.00	-4.50	0.00	8.00	4.50	5.75	1.25	-9.00	10.25	10.00	0.00	-3.50
W05	-3.75	-4.50	0.00	4.00	7.50	4.75	7.00	4.75	5.00	5.75	-7.75	4.50	5.00	0.00	-5.00
W06	2.50	-3.50	-5.25	-4.75	-10.50	3.50	-3.00	6.25	-4.00	-3.25	3.00	-3.00	-12.00	0.00	3.45
W07	3.50	-4.50	-5.25	-8.00	-7.00	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-12.50
W08	3.75	-6.25	-5.00	-4.50	-4.75	-6.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-2.25
W09	4.50	0.00	-0.25	-5.75	-5.00	4.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.75
W10	-3.25	-5.00	0.00	-1.25	-5.75	3.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-12.50
W11	3.50	-2.25	0.00	9.00	7.75	-3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4.50
W12	3.50	-2.00	-1.00	-10.25	-4.50	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-3.45
W13	1.25	5.75	-5.00	-10.00	-5.00	12.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	14.75

Le tableau augmenté est :

	Z01	Z02	Z03	Z04	Z05	Z06	Z07	Z08	Z09	Z10	Z11	Z12	Z13	Z00	e
W01	1.00	-2.00	4.50	-6.75	-4.50	-2.50	-3.50	-3.75	-4.50	3.25	-3.50	-3.50	-1.25	1.00	2.50
W02	3.50	-6.50	-5.00	10.50	-6.50	-9.25	4.50	6.25	0.00	5.00	2.25	2.00	-5.75	1.00	-3.75
W03	4.50	-3.75	-5.00	12.50	-4.50	-3.00	5.25	5.00	0.25	0.00	0.00	1.00	5.00	1.00	12.00
W04	-12.50	-4.50	0.00	0.00	-4.50	0.00	8.00	4.50	5.75	1.25	-9.00	10.25	10.00	1.00	-3.50
W05	-3.75	-4.50	0.00	4.00	7.50	4.75	7.00	4.75	5.00	5.75	-7.75	4.50	5.00	1.00	-5.00
W06	2.50	-3.50	-5.25	-4.75	-10.50	3.50	-3.00	6.25	-4.00	-3.25	3.00	-3.00	-12.00	1.00	3.45
W07	3.50	-4.50	-5.25	-8.00	-7.00	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	-12.50
W08	3.75	-6.25	-5.00	-4.50	-4.75	-6.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	-2.25
W09	4.50	0.00	-0.25	-5.75	-5.00	4.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	3.75
W10	-3.25	-5.00	0.00	-1.25	-5.75	3.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	-12.50
W11	3.50	-2.25	0.00	9.00	7.75	-3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	4.50
W12	3.50	-2.00	-1.00	-10.25	-4.50	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	-3.45
W13	1.25	5.75	-5.00	-10.00	-5.00	12.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	14.75

La colonne du pivot pour l'itération 1 est 14 (sa recherche est immédiate).

La ligne du pivot pour l'itération 1 est 7.

Le tableau de l'itération 1 est :

	Z01	Z02	Z03	Z04	Z05	Z06	Z07	Z08	Z09	Z10	Z11	Z12	Z13	W07	θ
W01	-2.50	2.50	9.75	1.25	2.50	-5.50	-3.50	-3.75	-4.50	3.25	-3.50	-3.50	-1.25	1.00	15.00
W02	0.00	-2.00	0.25	16.50	0.50	-12.25	4.50	6.25	0.00	5.00	2.25	2.00	-5.75	5.00	8.75
W03	1.00	0.75	0.25	20.50	2.50	-6.00	5.25	5.00	0.25	0.00	0.00	1.00	5.00	1.00	24.50
W04	-16.00	0.00	5.25	8.00	2.50	-5.00	8.00	4.50	5.75	1.25	-9.00	10.25	10.00	1.00	9.00
W05	-7.25	0.00	5.25	12.00	14.50	1.75	7.00	4.75	5.00	5.75	-7.75	4.50	5.00	1.00	7.50
W06	-1.00	1.00	0.00	3.25	-3.50	0.50	-3.00	6.25	-4.00	-3.25	3.00	-3.00	-12.00	1.00	15.95
Z00	-2.50	4.50	5.25	6.00	7.00	-3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	12.50
W08	0.25	-1.75	0.25	3.50	2.25	-9.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	10.25
W09	1.00	4.50	5.00	2.25	2.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	16.25
W10	-6.75	-0.50	5.25	6.75	1.25	0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
W11	0.00	2.25	5.25	17.00	14.75	-6.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	17.00
W12	0.00	2.50	4.25	-2.25	2.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	9.05
W13	-2.25	17.25	0.25	-2.00	2.00	9.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	27.25

La colonne du pivot pour l'itération 2 est 7.

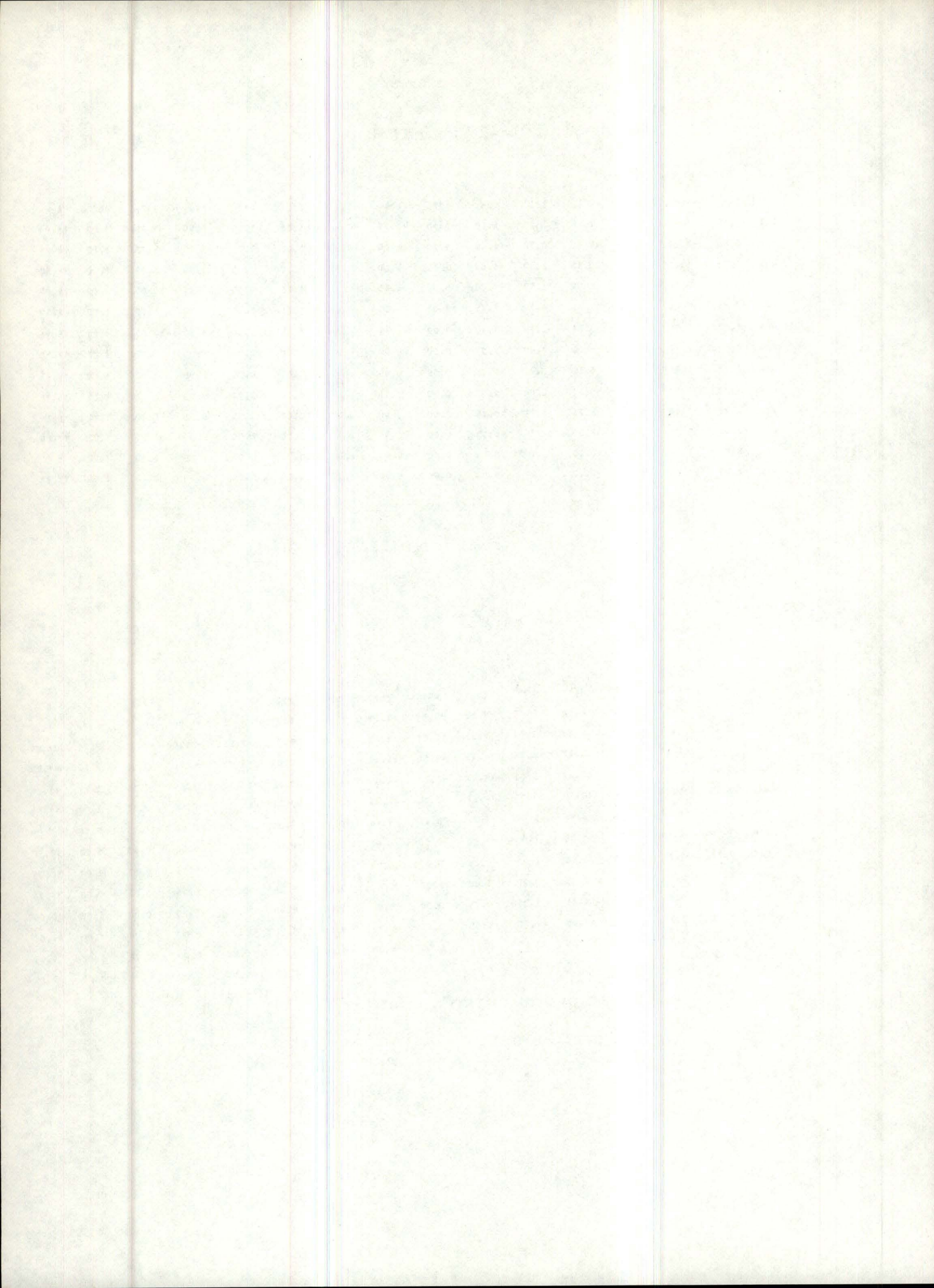
La ligne du pivot pour l'itération 2 est 1.

Le tableau de l'itération 2 est :

	Z01	Z02	Z03	Z04	Z05	Z06	W01	Z08	Z09	Z10	Z11	Z12	Z13	W07	θ
Z07	-0.71	0.71	2.79	0.36	0.71	-1.57	-0.29	-1.07	-1.29	0.93	-1.00	-1.00	-0.36	0.29	4.29
W02	-3.21	1.21	12.79	20.11	3.71	-10.32	-1.29	1.43	-5.79	9.18	-2.25	-2.50	-7.36	2.29	28.04
W03	-2.75	4.50	14.87	22.37	6.25	-14.25	-1.50	-0.62	-6.50	4.87	-5.25	-4.25	3.13	2.50	47.00
W04	-21.71	5.71	27.54	10.86	8.21	-15.57	-2.29	-4.37	-4.54	8.68	-17.00	2.25	7.14	3.29	43.29
W05	-12.25	5.00	24.75	14.50	19.50	-9.25	-2.00	-2.75	-4.00	12.25	-14.75	-2.50	2.50	3.00	37.50
W06	1.14	-1.14	-0.36	2.18	-5.64	5.21	0.86	9.44	-0.14	-6.04	6.00	0.00	-10.93	0.14	3.09
Z00	-3.50	4.50	5.25	6.00	7.00	-3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	12.50
W08	0.25	-1.75	0.25	3.50	2.25	-9.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	10.25
W09	1.00	4.50	5.00	2.25	2.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	16.25
W10	-6.75	-0.50	5.25	6.75	1.25	0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
W11	0.00	2.25	5.25	17.00	14.75	-6.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	17.00
W12	0.00	2.50	4.25	-2.25	2.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	9.05
W13	-2.25	17.25	0.25	-2.00	2.00	9.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	27.25

La colonne du pivot pour l'itération 3 est 1.

La ligne du pivot pour l'itération 3 est 10.



Le tableau de l'itération 3 est :

	Z07	Z02	Z03	Z04	Z05	Z06	Z01	Z08	Z09	Z10	Z11	Z12	Z13	WC7	0
Z07	0.11	0.77	2.23	-0.36	0.58	-1.60	-0.29	-1.07	-1.29	0.93	-1.00	-1.00	-0.36	0.18	4.29
W02	0.48	1.45	10.29	16.39	3.12	-19.44	-1.29	1.43	-5.79	9.18	-2.25	-2.50	-7.36	1.21	28.04
W03	0.41	4.70	12.74	19.62	5.74	-14.35	-1.50	-0.52	-6.50	4.87	-5.25	-4.25	3.13	2.09	47.00
W04	3.22	7.32	10.65	-10.86	4.19	-16.38	-2.29	-4.07	-4.54	6.68	-17.00	2.25	7.14	0.07	43.29
W05	1.81	5.91	15.26	2.25	17.23	-0.70	-2.00	-2.75	-4.00	12.25	-14.75	-2.50	2.50	1.19	37.50
W06	-0.17	-1.23	-7.47	3.32	-5.43	5.26	0.86	9.46	-0.14	-6.04	6.00	0.00	-10.93	0.31	3.09
Z00	0.52	4.76	2.53	4.50	6.15	-3.13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.48	12.50
W08	-0.04	-1.77	0.44	3.75	2.10	-9.24	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.04	10.25
W09	-0.15	4.43	5.78	3.25	2.19	1.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.15	16.25
Z01	-0.15	-0.07	0.78	1.00	0.19	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.15	0.00
W11	0.00	2.25	5.25	17.00	14.75	-6.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	17.00
W12	0.00	2.50	4.25	-2.25	2.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	9.00
W13	0.33	10.42	-1.50	-4.25	1.58	8.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.67	27.25

La colonne du pivot pour l'itération 4 est 10.

La ligne du pivot pour l'itération 4 est 6.

Le tableau de l'itération 4 est :

	Z02	Z03	Z04	Z05	Z06	Z01	Z08	Z09	Z06	Z11	Z12	Z13	WC7	0	
Z07	0.08	0.58	1.06	0.15	-0.25	-0.79	-0.15	0.38	-1.31	-0.15	-0.08	-1.00	-2.04	0.23	4.76
W02	0.22	-0.41	-1.07	11.94	-5.14	-11.45	0.02	15.82	-6.00	-1.52	6.87	-2.50	-23.98	2.28	32.74
W03	0.27	3.71	6.70	22.31	1.35	-11.11	-0.81	7.02	-6.62	-0.81	-0.40	-4.25	-5.70	2.34	49.50
W04	2.97	5.56	-0.09	-6.08	-3.62	-8.62	-1.05	9.54	-4.74	-1.44	-8.37	2.25	-2.57	0.52	47.73
W05	1.47	3.42	0.06	8.99	6.21	0.97	-0.26	16.46	-4.29	-2.03	-2.57	-2.50	-19.68	1.22	43.78
Z10	-0.03	-0.20	-1.24	0.55	-0.90	0.87	0.14	1.57	-0.02	-0.17	0.99	0.00	-1.81	0.05	0.51
Z00	0.52	4.76	2.53	4.50	6.15	-3.13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.48	12.50
W08	-0.04	-1.77	0.44	3.75	2.10	-9.24	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.04	10.25
W09	-0.15	4.43	5.78	3.25	2.19	1.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.15	16.25
Z01	-0.15	-0.07	0.78	1.00	0.19	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.15	0.00
W11	0.00	2.25	5.25	17.00	14.75	-6.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	17.00
W12	0.00	2.50	4.25	-2.25	2.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	9.00
W13	0.33	10.42	-1.50	-4.25	1.58	8.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.67	27.25

La colonne du pivot pour l'itération 5 est 6.

La ligne du pivot pour l'itération 5 est 8.

Le tableau de l'itération 5 est :

	W10	Z02	Z03	Z04	Z05	W08	W01	Z08	Z09	W06	Z11	Z12	Z13	W07	0
Z07	0.08	0.73	1.04	-0.17	-0.45	0.09	-0.15	0.38	-1.31	-0.15	-0.08	-1.00	-2.04	0.14	3.89
W02	0.26	1.78	-1.62	17.30	-7.98	1.24	0.02	15.82	-6.00	-1.52	6.87	-2.50	-23.98	1.00	20.04
W03	0.31	5.65	6.22	18.21	-1.16	1.09	-0.81	7.02	-6.62	-0.81	-0.40	-4.25	-5.70	1.21	32.29
W04	3.01	7.25	-0.52	-9.06	-5.81	0.95	-1.05	9.54	-4.74	-1.44	-8.37	2.25	-2.57	-0.47	37.95
W05	1.47	1.23	0.11	9.36	6.45	-0.16	-0.26	16.46	-4.29	-2.03	-2.57	-2.50	-19.62	1.93	44.25
Z10	-0.03	-0.37	-1.20	0.90	-0.68	-0.09	0.14	1.57	-0.02	-0.17	0.99	0.00	-1.81	0.15	1.48
Z00	0.53	5.36	2.32	3.23	5.57	0.34	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.13	9.03
Z06	-0.00	-0.19	0.05	0.41	0.25	-0.11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.11	1.11
W09	-0.15	4.23	5.63	3.67	2.44	-0.11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.26	17.40
Z01	-0.15	-0.08	0.78	1.62	0.19	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.15	0.04
W11	0.02	3.40	4.96	14.57	13.26	0.65	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	10.34
W12	0.00	2.50	4.25	-2.25	2.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	9.05
W13	0.30	4.71	-1.07	-0.63	3.80	-0.96	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.67	37.14

Les composantes du vecteur z_{08} sont toutes positives

La terminaison est anormale et le programme s'arrête dans l'indécision.

Remarques

- La démonstration de la convergence de l'algorithme de Lemke est basée sur des arguments de type combinatoire.
[15]

- Les classes de matrices pour lesquelles l'existence des solutions a été établie quand $z = (z_1 \dots z_n)^T$ sont $P[9]$, $SSM[11]$, $SCM[11]$, $ZP[2]$, PD dont les définitions des classes sont données ci-dessous

- L'algorithme de Lemke présente un inconvénient : l'arrêt possible dans l'indécision. Lemke a cependant montré que pour certaines classes de matrices, l'arrêt dans l'indécision signifie qu'il n'y a pas de solution réalisable au problème. Il en est ainsi pour les matrices co-positives plus (notation : $c+$), dont la définition

$$\text{est : } M \in C^+ \Leftrightarrow \left. \begin{array}{l} \text{a) } \forall z \quad z \geq 0 \Rightarrow z^T M z \geq 0 \\ \text{b) } \forall z \quad \left. \begin{array}{l} z \geq 0 \\ z^T M z = 0 \end{array} \right\} \Rightarrow (M + M^T) z = 0 \end{array} \right\} \text{ (co-positive CP)}$$

Exemples :

- a) Les matrices semi-définies positives (notation : PSD)

$$\forall z \quad z^T M z \geq 0$$

- b) Les matrices définies positives (notation : PD)

$$\forall z \neq 0 \quad z^T M z > 0$$

On peut énoncer la proposition suivante : si M est définie positive, alors tous les mineurs principaux de M sont strictement positifs. On a la proposition équivalente si M est symétrique.

- c) Toute combinaison linéaire à coefficients non-négatifs de $c+$ -matrices est une $c+$.

- d) Si M_1 et M_2 sont $c+$ et si A est quelconque, alors

$$\begin{pmatrix} M_1 & -A^T \\ A & M_2 \end{pmatrix} \text{ est } c+$$

Définitions :

- 1) La matrice M est strictement copositive (notation : SCP) si

$$\forall z > 0 \quad z^T M z > 0$$

- 2) La matrice M est une PO-matrice (notation : PO) si M a tous ses mineurs principaux positifs.

- 3) La matrice M est une Z-matrice (notation : Z) si M a ses éléments négatifs ou nuls, exceptés, éventuellement, ceux de la diagonale principale.

- 4) La matrice M est une ZP-matrice (notation : ZP) si M est à la fois une Z-matrice et une P-matrice.

Exemples : $M \equiv \begin{pmatrix} 2 & -5 \\ 1 & 2 \end{pmatrix} \Rightarrow M \in P \setminus \{PD, UZP\}$

$M \equiv \begin{pmatrix} 2 & -4 \\ 0 & 2 \end{pmatrix} \Rightarrow M \in ZP \setminus PD$

$M \equiv \begin{pmatrix} 1 & -4 \\ -4 & 1 \end{pmatrix} \Rightarrow M \in Z \setminus \{ZP, UZP\}$

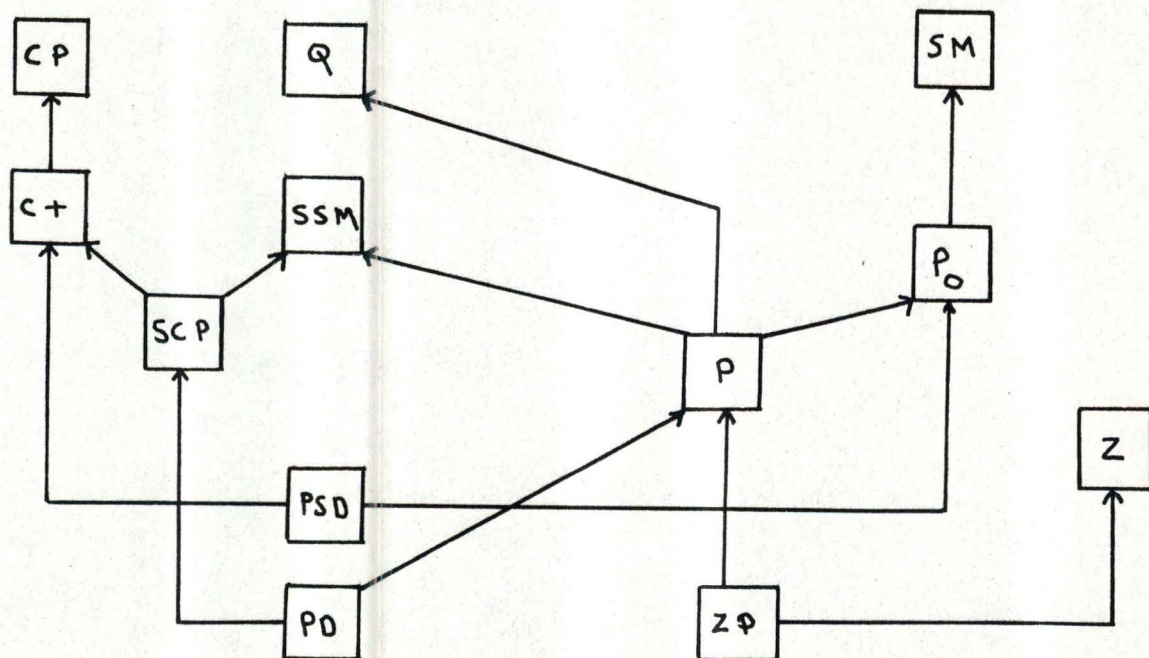
- 5) La matrice M est semi-monotone (notation : SM) si

$$\forall \alpha > 0 \quad \exists k \text{ tq } \alpha_k > 0 \text{ et } M_k \alpha \geq 0$$

La matrice M est strictement semi-monotone (notation : SSM) si $\forall \alpha \geq 0$
 $\alpha \neq 0 \quad \exists k \text{ tq } \alpha_k > 0 \text{ et } M_k \alpha > 0$

- 6) La matrice M est un Q-matrice si le LCP (q, M) a une solution pour chaque q appartenant à \mathbb{R}^n .

Les différentes classes de matrices qui viennent d'être définies peuvent être présentées sous forme d'un tableau-résumé. Les plus petites d'entre elles se situent à la base de ce tableau.



Algorithme de Cottle et Dantzig

Plutôt que de décrire l'algorithme de Cottle et Dantzig dans tous ses détails, nous nous attacherons ici à sa "philosophie".

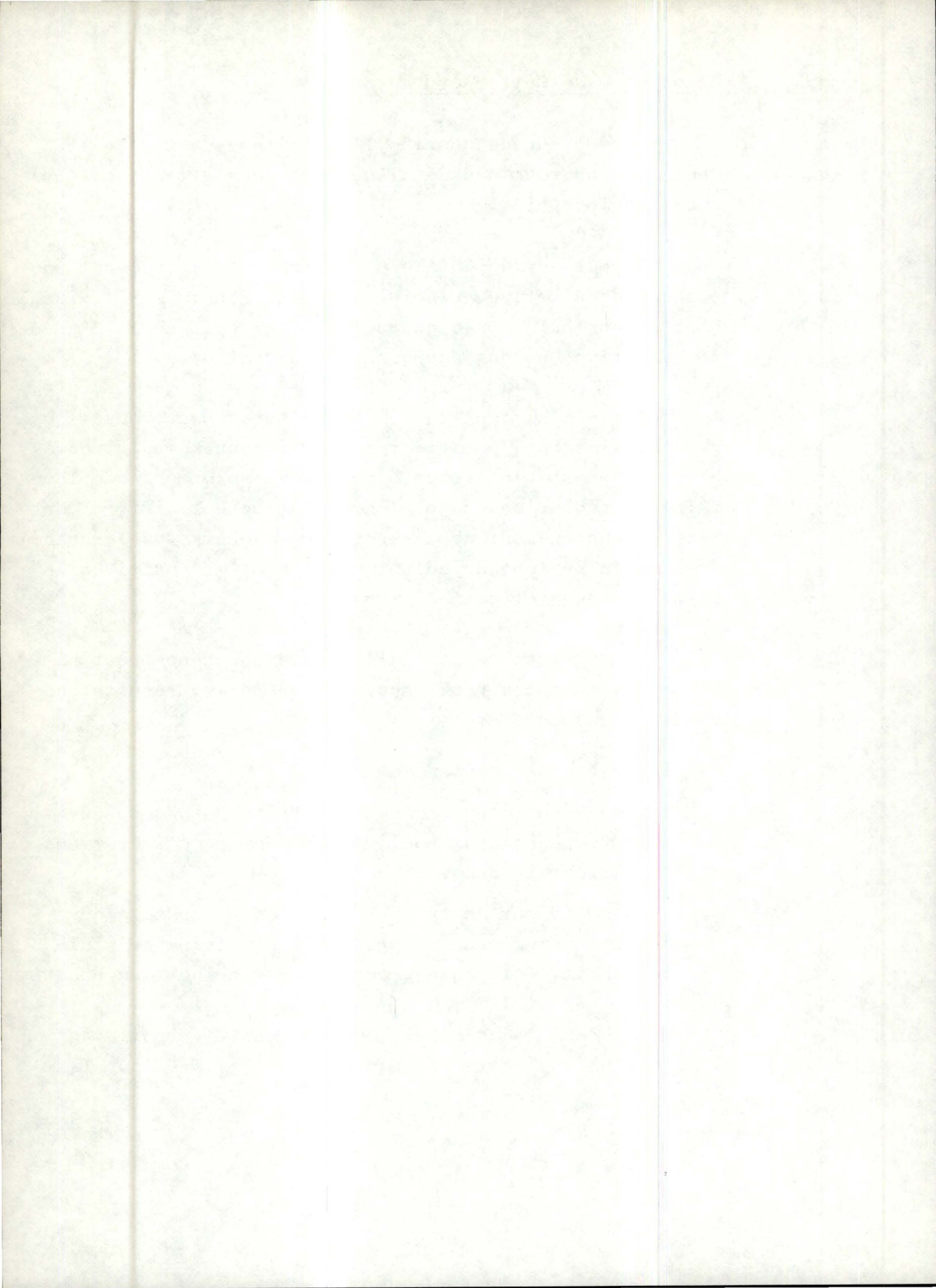
La méthode de Cottle et Dantzig [5]; [6] est une méthode de pivotage principal (définition : voir lexique). Elle a pour but, après chaque pivotage principal, de rendre positive une des composantes du vecteur q .

Toutefois, tout au long des itérations, toutes les variables de base positives sont maintenues positives. Le cheminement s'effectue donc ici de solutions complémentaires en solutions complémentaires, ce qui diffère fortement du cheminement de l'algorithme de Lemke où les pas successifs se font de solutions augmentées positives en solutions augmentées positives.

La convergence de l'algorithme en un nombre fini d'étapes, pour les problèmes non-dégénérés, résulte des conditions suivantes :

- 1) le nombre de cycles est fini;
- 2) si, au début d'un cycle, la matrice M possède des mineurs principaux positifs, le nombre d'itérations à l'intérieur du cycle est fini;
- 3) à chaque début de cycle la matrice M^T possède des mineurs principaux positifs si la matrice initiale M possédait, elle aussi, des mineurs principaux positifs.

Eu égard à ces conditions, on vérifie facilement que l'algorithme de Cottle et Dantzig est vérifié pour la classe des matrices PSD.



C H A P I T R E I I

- - - - -
- - - - -

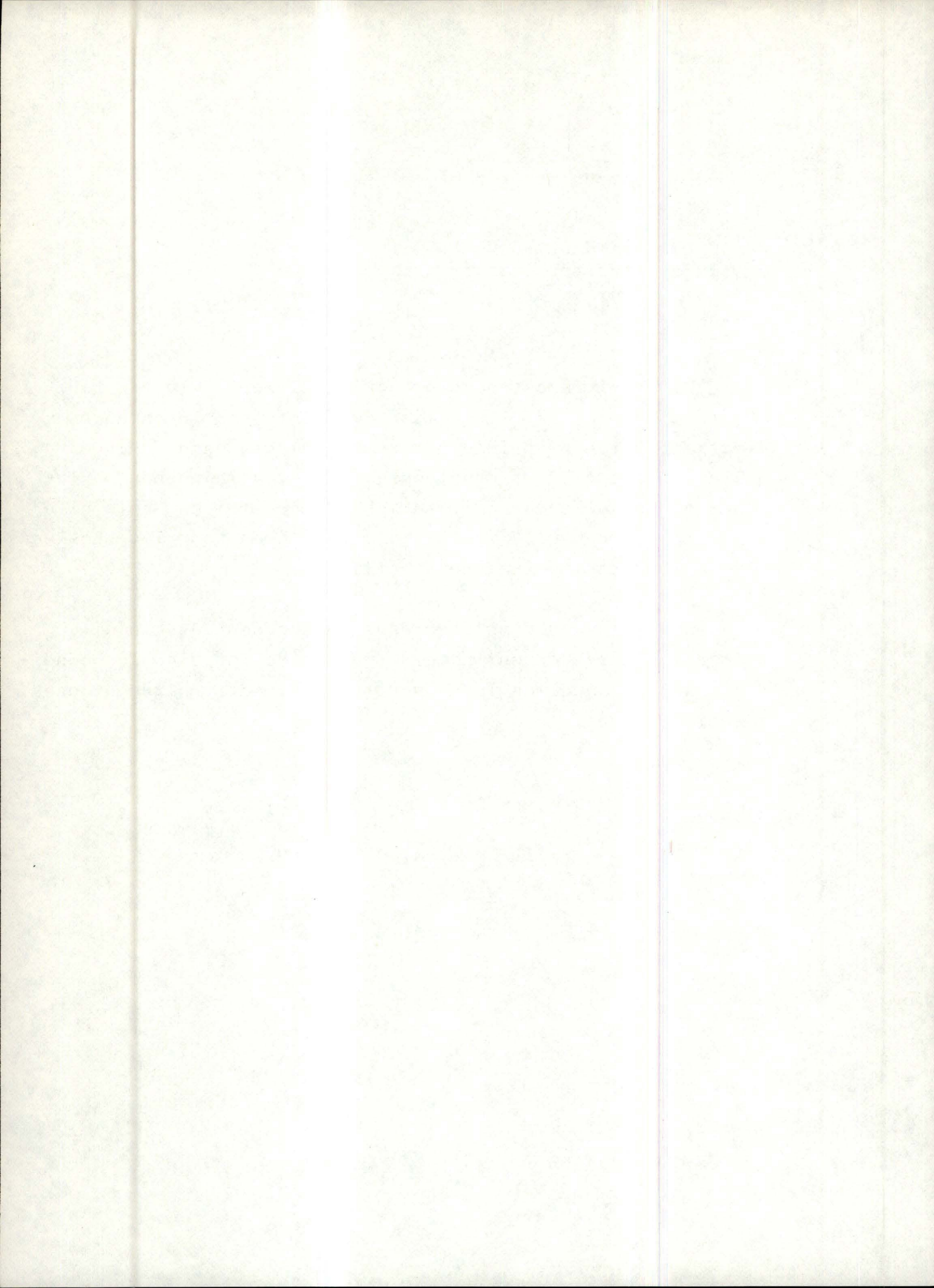
CARACTERISATION DE LA CLASSE P

=====

INTRODUCTION

Ce chapitre est consacré principalement à la propriété du "trou du pigeon". La notion de points complémentaires vérifiant cette propriété est probablement, dans ce mémoire, celle qui est la plus fructueuse. Elle entraîne non seulement une nouvelle caractérisation des P-matrices, mais elle engendre une nouvelle famille d'algorithmes pour les problèmes de complémentarité (voir chap. III).

Nous l'évoquerons donc dans chacun des cas linéaire et non-linéaire, ce qui nous permettra de formuler les conditions sous lesquelles les algorithmes directs possèdent une solution unique.



1 DEFINITIONS IMPORTANTES

Afin de clarifier la présentation et d'unifier les notations, nous précisons en ce début du chapitre qui nous amène au coeur du problème, quelques définitions importantes.

Définition 2.1.

Soit une fonction continue on définit un problème de complémentarité associé à la fonction f (que l'on notera par cpf) comme suit :

chercher un vecteur x de \mathbb{R}^m tel que, pour chaque indice i appartenant à l'ensemble N , on ait :

$$(1) \quad x_i \geq 0$$

$$(2) \quad f_i(x) \geq 0$$

$$(3) \quad x_i f_i(x) = 0$$

La condition (3) du cpf est similaire à la condition de complémentarité.

Définition 2.2.

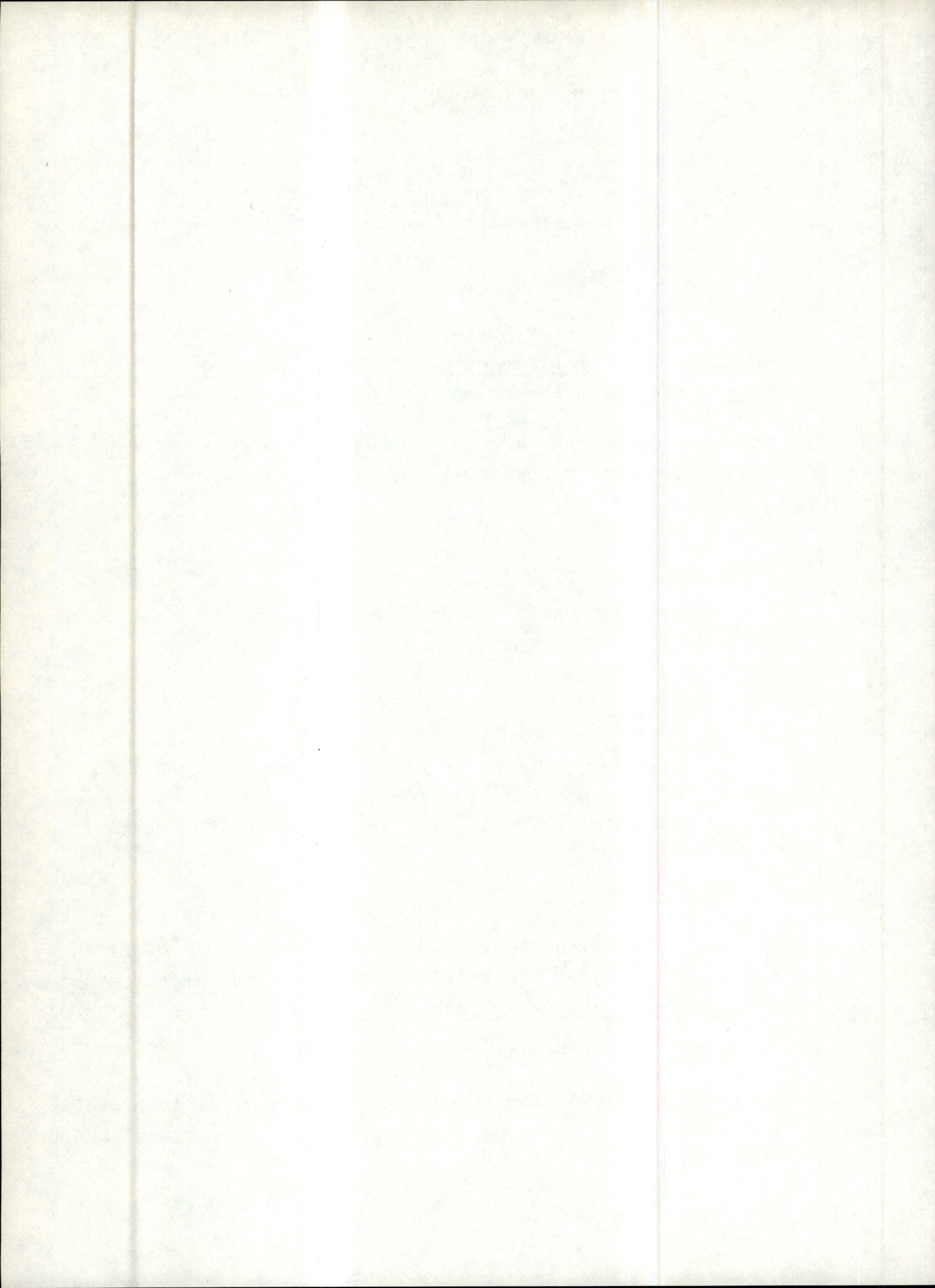
Si $y = f(x)$, alors le cpf est non-dégénéré si, pour chaque x de l'espace \mathbb{R}^m solution de cpf, il y a au plus n , des $2n$ variables (y, x) , qui s'annulent simultanément.

Définition 2.3.

Supposons que le vecteur z appartenant à \mathbb{R}^m vérifie la condition de complémentarité (3) du cpf. On définit alors le point complémentaire " s " associé au vecteur z par la formule

$$s = z + f(z)$$

Cette définition nous montre donc que, si le problème considéré est non-dégénéré, le vecteur s ne contient aucune composante nulle.



Nous supposons dans la suite de ce mémoire que les problèmes traités sont non-dégénérés.

Dans cette partie, nous porterons un intérêt tout particulier à la condition de complémentarité; nous laisserons momentanément de côté les conditions (1) et (2) du cpf.

Définition 2.4.

On notera par \underline{N} l'ensemble des entiers positifs de 1 à n . $N \triangleq \{1, 2, 3, \dots, m-1, m\}$

Définition 2.5.

On notera par \underline{X} l'ensemble des points complémentaires du problème de complémentarité associé à la fonction f .

Donc,

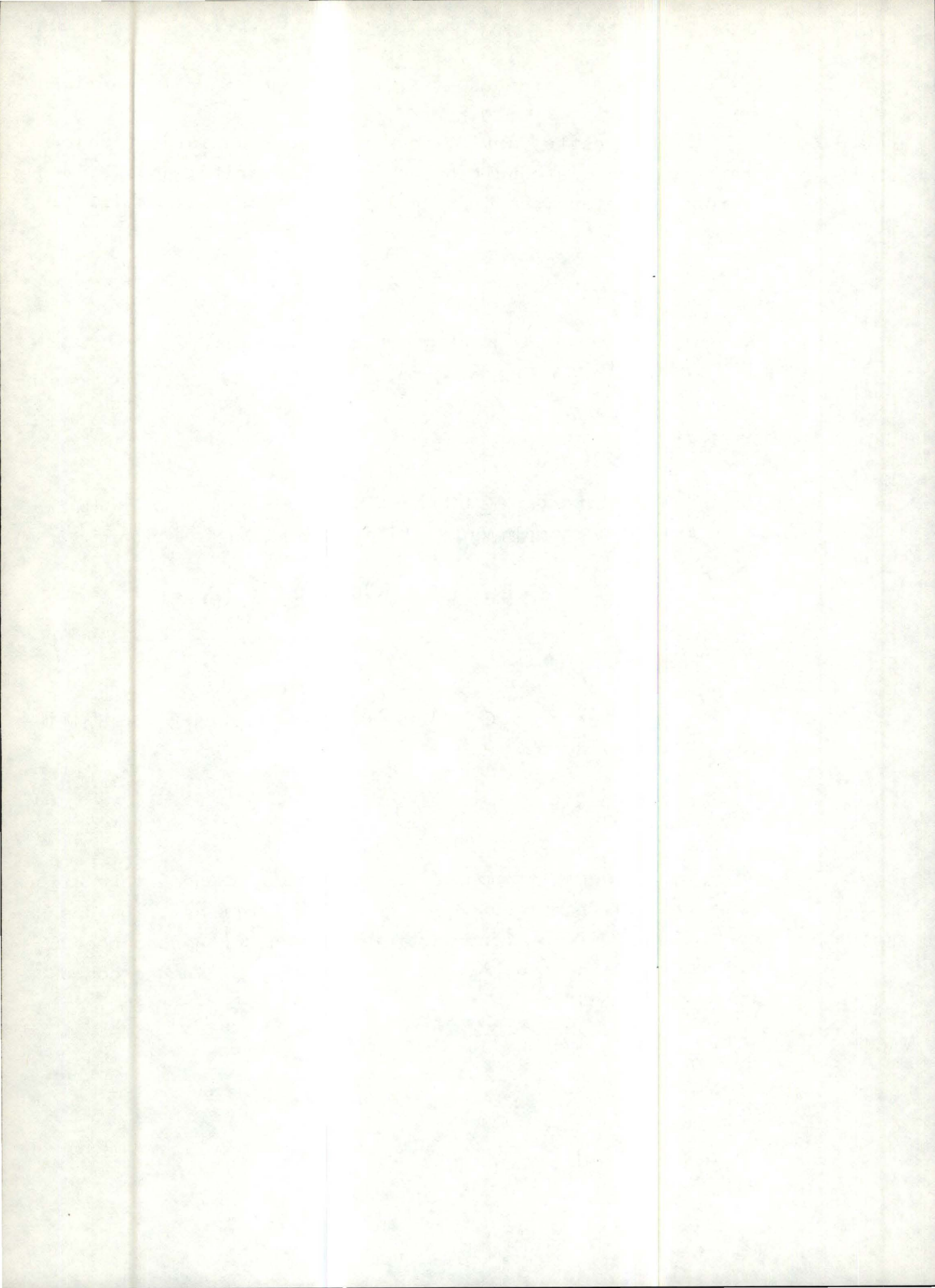
$$\underline{X} = \{ \lambda = z + f(z) \mid z \in \mathbb{R}^m \text{ et } z_i \cdot f_i(z) = 0 \}$$

Définition 2.6.

On notera par \textcircled{W} l'ensemble des orthants (définition voir lexique) de \mathbb{R}^m .

Lemme 2.1.

Si le cpf est non dégénéré, alors \underline{X} a une intersection vide avec la frontière de n'importe quel orthant. La démonstration est évidente en raison de la non-dégénérescence du cpf qui entraîne : $\lambda \in \underline{X}$ ce qui entraîne qu'aucune composante de λ n'est nulle.



2 POSITION DU PROBLEME

Notre but est d'arriver à la "propriété du trou du pigeon" que nous énoncerons comme suit :

Chaque orthant de \mathbb{R}^m contient un et un seul point complémentaire. Pour cela, nous montrerons que sous certaines conditions, il existe :

1) 2^n points complémentaires;

2) au moins un point complémentaire par orthant.

Dès lors, comme il y a 2^n orthants dans l'espace \mathbb{R}^m , le principe du "trou du pigeon", énoncé ci-dessous, exige qu'il y ait un et un seul point complémentaire par orthant.

Principe du "trou du pigeon"

Si m pigeons doivent nicher dans m trous et que l'on ne peut placer qu'un pigeon au maximum par trou, tous les trous seront occupés chacun par un et un seul pigeon. Ce principe du "trou du pigeon" constitue une interprétation imagée du résultat suivant :

Soient deux ensembles A et B , finis, distincts, de même cardinalité.

Soit g une correspondance entre ces deux ensembles

$$g: A \rightarrow B; \#A = \#B \text{ fini}$$

Si cette correspondance g est fonctionnelle,
partout définie et
surjective,

alors elle est injective et donc bijective.

La première assertion : " X possède 2^n éléments" se montrera par l'établissement d'une bijection S .

$$S: \mathcal{P}(N) \rightarrow X$$

En établissant d'autre part le caractère surjectif d'une application : $g: X \rightarrow \oplus$

nous montrerons le second point.

3 DEFINITION ET DEMONSTRATION DU CARACTERE BIJECTIF DE SA Définition de la correspondance S

Soit S une correspondance entre les ensembles $\mathcal{P}(N)$ et \mathcal{X}

$$S: \mathcal{P}(N) \longrightarrow \mathcal{X}$$

définie de la manière suivante :

Etant donné $I \in \mathcal{P}(N)$, on associe à I les points complémentaires éventuels correspondant aux solutions, si elles existent, du problème :

$$\begin{cases} z_i = 0 & \forall i \in N \setminus I \\ f_i(z) = 0 & \forall i \in I \end{cases} \quad (a)$$

Exemple :

1) Soient $N = \{1, 2, 3\}$

$$f: \mathbb{R}^3 \longrightarrow \mathbb{R}^3$$

$$(\alpha_1, \alpha_2, \alpha_3) \rightsquigarrow (\ln(\alpha_1 - \alpha_2), \ln(\alpha_1 + \alpha_2), \ln \alpha_3)$$

$$I = \{1\}$$

déterminons, à partir des formules (a), le système à résoudre

$$\begin{array}{l} 1 \in I \\ 2 \notin I \\ 3 \notin I \end{array} \Rightarrow \begin{cases} f_1(\alpha) = 0 \\ \alpha_2 = 0 \\ \alpha_3 = 0 \end{cases}$$

Il s'écrit donc

$$\begin{cases} \ln(\alpha_1 - \alpha_2) = 0 \\ \alpha_2 = 0 \\ \alpha_3 = 0 \end{cases} \iff \begin{cases} \alpha_1 - \alpha_2 = 1 \\ \alpha_2 = 0 \\ \alpha_3 = 0 \end{cases} \iff \begin{cases} \alpha_1 = 1 \\ \alpha_2 = 0 \\ \alpha_3 = 0 \end{cases}$$

Le point complémentaire Λ associé à l'ensemble d'indexation I est :

$$\Lambda = (1, 0, 0)$$

2) Soient $N = \{1, 2, 3, 4\}$

$$f: \mathbb{R}^4 \longrightarrow \mathbb{R}^4$$

$$(\alpha_1, \alpha_2, \alpha_3, \alpha_4) \mapsto (5\alpha_1, \alpha_2^2, \alpha_1 - \alpha_2, 1)$$

$$I = \{2, 4\}$$

Déterminons le système à résoudre :

$$\begin{array}{l} 1 \notin I \\ 2 \in I \\ 3 \notin I \\ 4 \in I \end{array} \Rightarrow \begin{cases} \alpha_1 = 0 \\ f_2(\alpha) = 0 \\ \alpha_3 = 0 \\ f_4(\alpha) = 0 \end{cases} \iff \begin{cases} \alpha_1 = 0 \\ \alpha_2^2 = 0 \\ \alpha_3 = 0 \\ 1 = 0 \end{cases}$$

Ce système est impossible à résoudre.

Nous ne pouvons donc pas associer à I des points complémentaires. Si l'on fait une analogie avec la terminologie de programmation linéaire, on remarque que le vecteur Δ est décrit comme un vecteur de base. En effet, il contient les variables

$$\begin{cases} \Delta_i = z_i & \forall i \in I_z \\ \Delta_i = f_i(z) & \forall i \in N \setminus I_z \end{cases}$$

et les variables $(z_i, f_i(z))$ qui n'appartiennent pas à ce vecteur Δ s'annulent (variables hors-base).

B Le caractère surjectif de S : $\mathcal{P}(N) \longrightarrow \mathcal{X}$

Le caractère surjectif de S s'obtient à partir de la définition même du point complémentaire. En effet, par (3), il existe un ensemble d'indexation I_Δ inclus dans N tel que

$$\begin{cases} f_i(z) = 0 & \forall i \in I_\Delta \\ z_i = 0 & \forall i \in N \setminus I_\Delta \end{cases} \quad (2.1)$$

$$\text{On a donc : } \begin{cases} \lambda_i = z_i & \forall i \in I_\lambda \\ \lambda_i = f_i(z) & \forall i \in N \setminus I_\lambda \end{cases}$$

La résolution du système (2.1) établit le caractère surjectif de la correspondance S .

C La correspondance $S : \mathcal{P}(N) \longrightarrow \mathcal{X}$ est partout définie et fonctionnelle.

Montrer le caractère "partout défini" de la correspondance S reviendra à démontrer, dans le théorème 2.2, l'existence d'un point complémentaire pour chaque ensemble d'indexation. Démontrer l'unicité de ce point complémentaire associé à un ensemble d'indexation permettra alors de déduire le caractère fonctionnel. La correspondance S sera donc une application.

Définition 2.7.

Si f est une fonction continue de l'espace \mathbb{R}^m sur lui-même, alors elle est une P-fonction sur l'ensemble $T \in \mathbb{R}^m$ si pour tout x, y différent appartenant à T , il existe un indice i tel que

$$(x_i - y_i) (f_i(x) - f_i(y)) > 0$$

Une classe importante de P-fonctions est décrite par le théorème suivant.

Théorème 2.1.

Supposons que la fonction f soit différentiable sur \mathbb{R}^m . Si $J_f(x)$, la matrice jacobienne pour $f(x)$, est une P-matrice sur \mathbb{R}^m , alors $f(x)$ est une P-fonction sur \mathbb{R}^m .

Pour la démonstration, voir [18].

Définition 2.8.

Une fonction $f : \mathbb{R}^m \rightarrow \mathbb{R}^m$ possède une matrice jacobienne $J_f(x)$ bornée positivement (notation : $f \in \text{PBJM}$) si il existe un nombre δ compris entre 0 et 1 tel que pour chaque x de \mathbb{R}^m , tous les mineurs principaux de $J_f(x)$ sont compris entre δ et δ^{-1} .

Exemple Soient $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \mapsto \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} \sin x_1 + 2x_1 \\ x_1 + x_2 \end{pmatrix}$$

 $\delta = 1/4$

Cette fonction appartient à la classe PBJM. En effet, sa matrice jacobienne $J_f(x)$ est bornée positivement.

$$J_f(x) = \begin{pmatrix} \cos x_1 + 2 & 0 \\ 1 & 1 \end{pmatrix}$$

Lemme 2.2.

Soit $f : \mathbb{R} \rightarrow \mathbb{R}$ une fonction différentiable; supposons que $f'(x) > \delta > 0 \quad \forall x \in \mathbb{R}$

alors f est surjective.

Démonstration : elle se fait par l'absurde.

La fonction f est continue et strictement croissante, donc $f(\mathbb{R}) = I$ où I est un intervalle de \mathbb{R} .

Montrer que la fonction f est surjective revient à montrer que $I = \mathbb{R}$.

1) Supposons

$$f(x) \leq M \quad \forall x \in \mathbb{R} \quad \text{avec } M \text{ fini;}$$

alors, nous avons

$$\forall x < 0 \quad M - f(0) \leq f(x) - f(0) = x f'(\theta x) < x \delta \quad 0 < \theta < 1$$

Donc, en choisissant M suffisamment grand, on a l'absurdité suivante :

$$M - f(0) < \alpha \delta \quad \forall \alpha < 0$$

2) Supposons

$$f(x) \leq M \quad \forall x \in \mathbb{R} \quad \text{avec } M \text{ fini;}$$

alors, nous avons

$$\forall \alpha > 0 \quad M - f(0) \geq f(x) - f(0) = \alpha f'(\theta \alpha) > \alpha \delta \quad 0 < \theta < 1$$

Donc, en choisissant M suffisamment petit, on a l'absurdité suivante :

$$M - f(0) > \alpha \delta \quad \forall \alpha > 0$$

Ainsi donc, I est un intervalle non borné.

Théorème 2.2.

Supposons que la fonction f appartienne à PBJM et que le cpf correspondant soit non-dégénéré. Alors, à chaque ensemble d'indexation I^0 , on associe un et un seul point complémentaire λ .

Démonstration :

1) Existence du point complémentaire λ .

Considérons l'ensemble d'indexation I^0

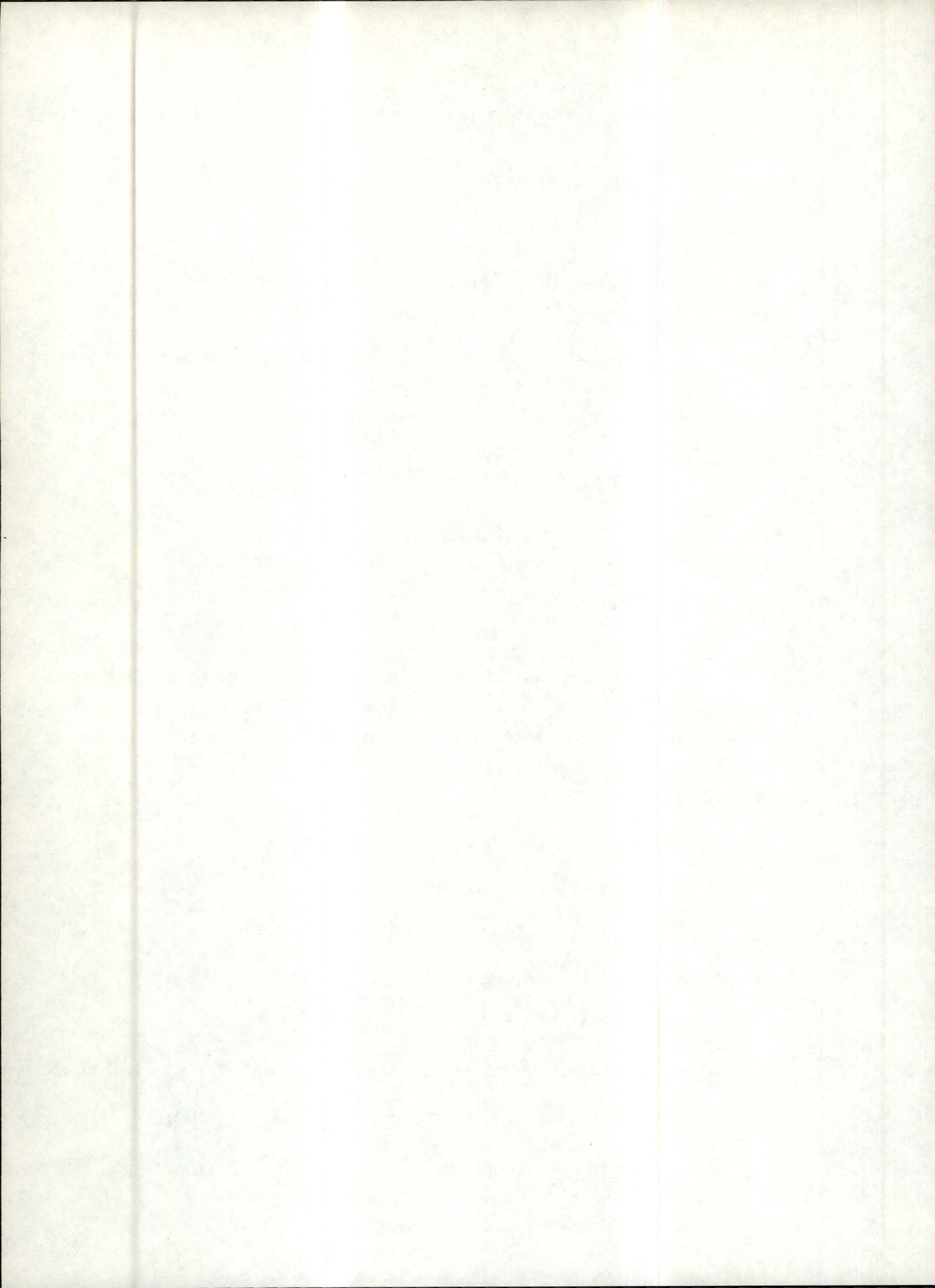
$$I^0 = \{ \lambda_1, \lambda_2, \dots, \lambda_n \} \quad \text{avec } n \in \mathbb{N}$$

l'ensemble des équations

$$y_i - f_i(x) = 0$$

le point complémentaire λ correspondant à $\begin{cases} x_i = 0 & \forall i \in \mathbb{N} \\ y_i = f_i(0) \end{cases}$

$$\lambda = 0 + f(0)$$



Nous allons échanger les rôles des variables η_{i_2} et κ_{i_2}
 Par hypothèse, nous avons :

$$f \in \text{PBJM}$$

ce qui entraîne

$$\exists \delta \in \mathbb{R} \text{ tel que } \left(\frac{\partial f_{i_2}}{\partial \kappa_{i_2}} \right) (\kappa) \geq \delta > 0 \quad \forall \kappa \in \mathbb{R}$$

Pour tout $(\kappa_1, \dots, \kappa_{i_2-1}, \kappa_{i_2+1}, \dots, \kappa_m)$ on définit
 l'application

$$f_{i_2, \kappa_1, \dots, \kappa_{i_2-1}, \kappa_{i_2+1}, \dots, \kappa_m} : \mathbb{R} \rightarrow \mathbb{R}$$

$$\kappa_{i_2} \rightsquigarrow f_{i_2}(\kappa_1, \dots, \kappa_{i_2}, \dots, \kappa_m)$$

La dérivée première étant positive, cette application est
 strictement croissante, donc injective.

Et comme sa dérivée est strictement positive, elle est
 surjective, donc bijective (lemme 2.2.).

Dès lors, cette application admet une application réciproque.

$$f_{i_2, \kappa_1, \dots, \kappa_{i_2-1}, \kappa_{i_2+1}, \dots, \kappa_m} : \mathbb{R} \rightarrow \mathbb{R}$$

$$\eta_{i_2} \rightsquigarrow \kappa_{i_2}$$

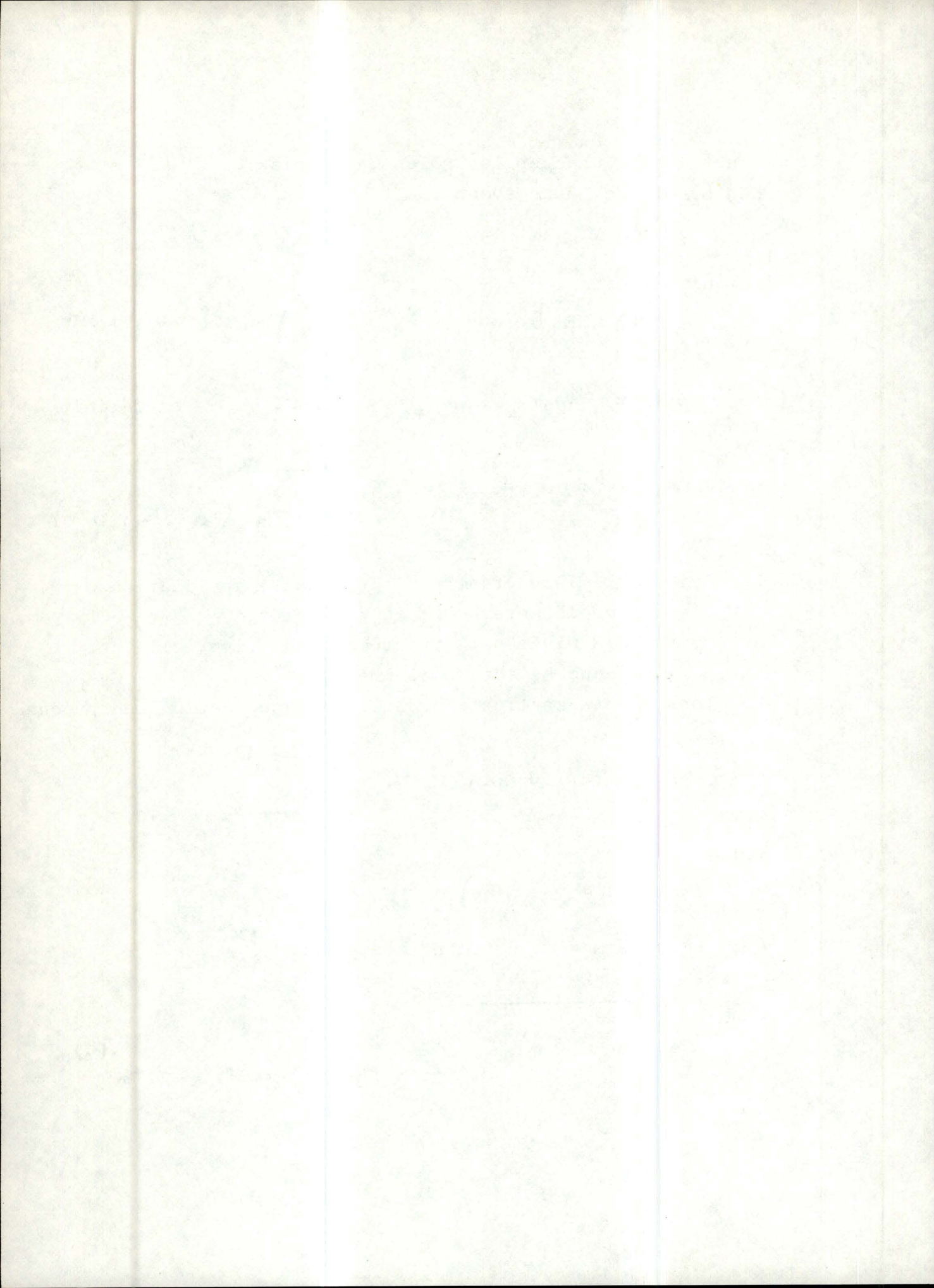
telle que

$$f_{i_2}(\kappa_1, \dots, \kappa_{i_2}, \dots, \kappa_m) = \eta_{i_2}$$

Ceci nous permet de définir la fonction $f_{i_2}^*$

$$f_{i_2}^* : \mathbb{R} \longrightarrow \mathbb{R}$$

$$(\kappa_1, \dots, \eta_{i_2}, \dots, \kappa_m) \rightsquigarrow f_{i_2}(\kappa_1, \dots, \kappa_{i_2-1}, \kappa_{i_2+1}, \dots, \kappa_m) \quad (\eta_{i_2})$$



On obtient alors un nouvel ensemble d'équations

$$y_i^* - f_i^*(x^*) = 0$$

avec

$$\begin{cases} y_{i_2}^* = x_{i_2} & \text{et } x_{i_2}^* = y_{i_2} \\ y_i^* = y_i & \text{et } x_i^* = x_i \quad \forall i \in N \setminus \{i_2\} \\ f_i^* = f_i(x_2^*, \dots, x_{i_2-1}^*, f_{i_2}^*(x^*), x_{i_2+1}^*, \dots, x_m^*) \quad \forall i \in N \setminus \{i_2\} \end{cases}$$

Si on échange seulement la dépendance des rôles de y_{i_2} et de x_{i_2} et, comme la condition (3) de cpf est symétrique en x_i et $f_i(x)$ pour tout indice i de N , l'ensemble des points complémentaires de la fonction f^* contient les points complémentaires de la fonction f .

Cottle [5] a prouvé que si f a une matrice jacobienne bornée positivement, alors f^* appartient également à PBJM.

Si nous appliquons le même raisonnement en sens inverse, en échangeant la dépendance des rôles de $y_{i_2}^*$ et de $x_{i_2}^*$, nous pouvons conclure que l'ensemble des points complémentaires de f contient les points complémentaires de f^* .

Donc, f et f^* possèdent les mêmes points complémentaires.

Si la fonction f^* appartient à PBJM, on peut effectuer un autre changement : échanger les rôles de $y_{i_2}^* = y_{i_2}$ et de $x_{i_2}^* = x_{i_2}$. C'est pourquoi, si l'ensemble d'indexation I^0 possède n éléments, on peut décomposer l'échange des variables correspondantes à I^0 en n échanges simples sur des fonctions PBJM.

Le point complémentaire Λ s'obtient par la transformation finale $f^{***} \dots^*$. C'est pourquoi

$$\Lambda = f^{***} \dots^* (0^{***} \dots^*)$$

2) Unicité du point complémentaire Λ

Supposons qu'il existe deux points complémentaires de f pour l'ensemble d'indexation I^0 .

Soient $\lambda = x + f(x)$ et $t = z + f(z)$, ces deux points.

Pour tout indice i de N , considérons la quantité

$$c_i = (x_i - z_i) (f_i(x) - f_i(z)) \quad (2.2)$$

On a donc :

$$c_i = \begin{cases} 0 & \text{si } i \in I^0 & \text{car } f_i(x) = 0 = f_i(z) \\ 0 & \text{si } i \notin I^0 & \text{car } x_i = 0 = z_i \end{cases} \quad (2.3)$$

De plus, comme $f \in \text{PBJM}$ entraîne que $J_f(x)$ est une P-matrice ce qui entraîne que f est une P-fonction, les relations (2.2) et (2.3) contredisent le théorème 2.1, puisque $\max_i c_i$ n'est pas strictement positif. On a donc $\rho = \tau$ et il y a au plus un point complémentaire associé à l'ensemble d'indexation I^0 .

Exemple :

Cet exemple montre l'importance dans le théorème 2.2 du nombre strictement positif intervenant dans la propriété PBJM.

Soit le système

$$\begin{cases} y_1 = f_1(x) = e^{x_2 - x_2} \\ y_2 = f_2(x) = e^{x_2} + x_2 \end{cases}$$

La matrice jacobienne associée à ce système est :

$$\begin{pmatrix} e^{x_2 - x_2} & -e^{x_2 - x_2} \\ e^{x_2} & 1 \end{pmatrix}$$

Ces mineurs principaux sont tous positifs.

Soit $I = \emptyset$, alors

$$\begin{cases} x_1 = x_2 = 0 \\ y_1 = y_2 = 1 \end{cases}$$

Soit $I = \{1\}$, alors

$$\begin{cases} x_1 = \ln y_1 + x_2 \\ y_2 = y_1 e^{x_2} + x_2 \end{cases}$$

En annulant les composantes x_2 et y_1 , on obtient :

$$\begin{cases} x_1 = -\infty \\ y_2 = 0 \end{cases}$$

Ces "solutions" sont étonnantes, mais on peut s'y attendre, puisqu'une hypothèse du théorème n'est pas vérifiée. En effet on n'a pas de véritable réciproque puisqu'il n'existe pas de nombre $0 < \delta < 1$ tel que les mineurs principaux de la matrice jacobienne soient tous compris entre δ et δ^{-1}

Conclusion :

On peut, à présent, affirmer que la correspondance $S: \mathcal{P}(N) \rightarrow \mathcal{X}$ est une application surjective.

D Le caractère injectif de l'application $S: \mathcal{P}(N) \rightarrow \mathcal{X}$

Ce caractère injectif découle du théorème suivant :

Théorème 2.3.

Si la fonction f est une P-fonction sur \mathbb{R}^m et si le cpf est non-dégénéré, alors deux ensembles d'indexation distincts ne peuvent donner lieu aux mêmes points complémentaires.

Démonstration : elle se fait par contraposition.

Considérons deux points complémentaires α et β égaux.

$$\alpha + f(\alpha) = \beta = \gamma + f(\gamma)$$

Par hypothèse, on a :

$$I_\alpha \neq I_\gamma$$

Donc, pour chaque indice i de N , on a la condition de complémentarité

$$\begin{aligned} (\alpha_i - \gamma_i)(f_i(\alpha) - f_i(\gamma)) &= \alpha_i f_i(\alpha) - \alpha_i f_i(\gamma) - \gamma_i f_i(\alpha) + \gamma_i f_i(\gamma) \\ &= -\alpha_i f_i(\gamma) - \gamma_i f_i(\alpha) \\ &\leq 0 \end{aligned}$$

La dernière inégalité provient du fait que les points λ et ϵ étant égaux, appartiennent au même orthant \textcircled{D}

Ce fait contredit l'hypothèse : "f est une P-fonction sur \mathbb{R}^m ", excepté si α et η sont égaux.

D'autre part la non-dégénérescence du cpf implique les relations suivantes :

$$\alpha_i = 0$$

$$\forall i \in N \setminus I_\alpha$$

$$\alpha_i \neq 0$$

$$\forall i \in I_\alpha$$

$$\eta_i = 0$$

$$\forall i \in N \setminus I_\eta$$

$$\eta_i \neq 0$$

$$\forall i \in I_\eta$$

Comme $\alpha = \eta$, on a :

$$I_\alpha = I_\eta$$

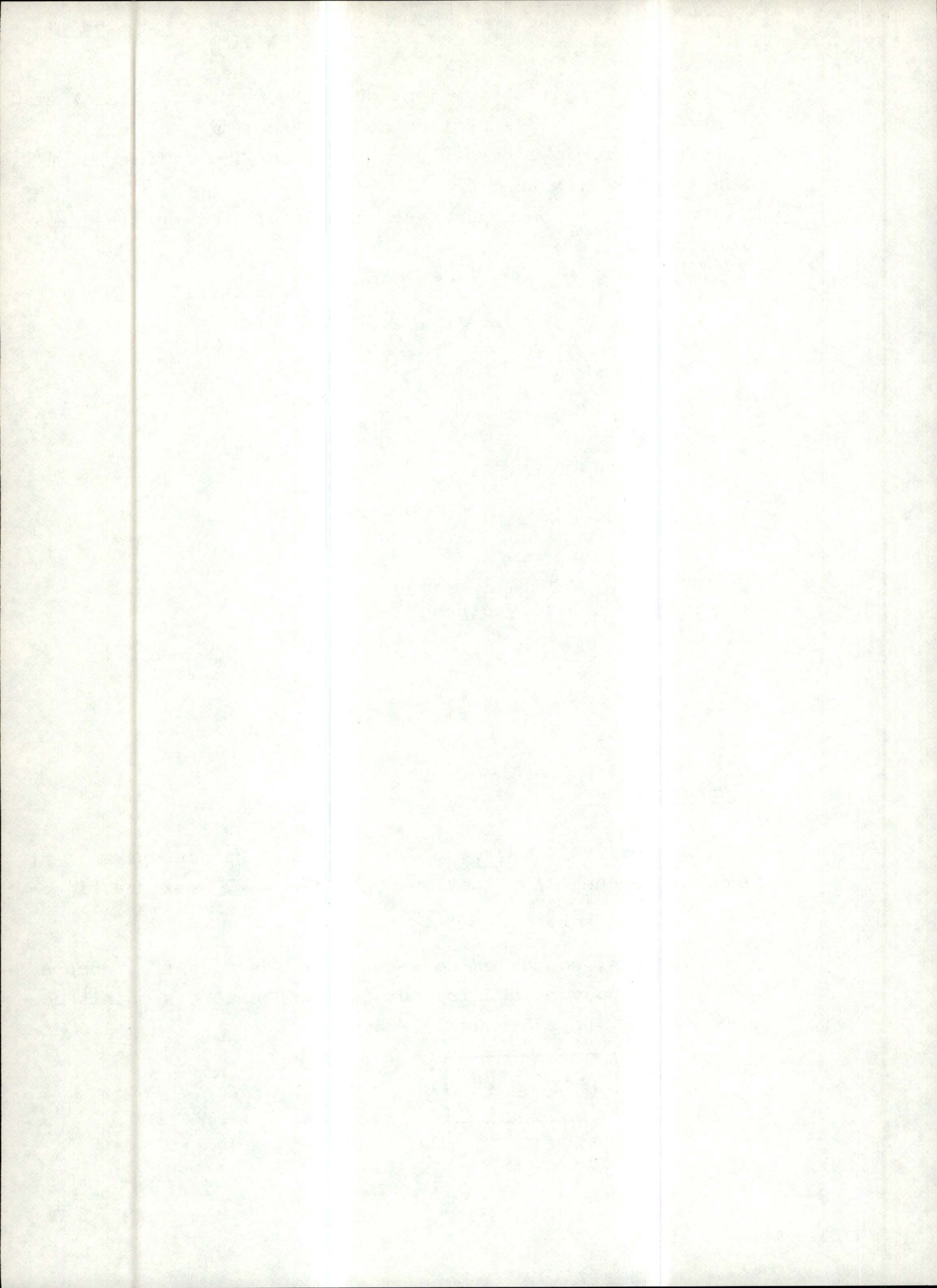
ce qui contredit l'hypothèse.

E Conclusions

Les points A, B, C, D de ce paragraphe nous permettent de conclure que l'application $S : \mathcal{P}(N) \rightarrow \mathcal{X}$ est une bijection.

De plus, comme les sous-ensembles de N sont au nombre de 2^n , nous pouvons déduire que \mathcal{X} , l'ensemble des points complémentaires, possède aussi 2^n éléments.

$$\boxed{\# \mathcal{X} = 2^m}$$



4 DEFINITION ET DEMONSTRATION DU CARACTERE SURJECTIF DE gA Définition de l'application g

Soit g la correspondance entre les ensembles \mathcal{X} et \mathbb{O}

$$g: \mathcal{X} \longrightarrow \mathbb{O}$$

définie de la manière suivante :

Tout point complémentaire d'un cpf non-dégénéré, possède une certaine configuration de signes à laquelle nous pouvons associer un certain orthant. Ainsi donc, à chaque point complémentaire correspond un orthant de \mathbb{R}^m . Ce fait nous permet alors de conclure que la correspondance g est partout définie et fonctionnelle.

Nous sommes donc en présence d'une application $g: \mathcal{X} \rightarrow \mathbb{O}$

B Le caractère surjectif de l'application g

Le caractère surjectif se démontre facilement à l'aide du théorème suivant :

Théorème 2.4

Si la fonction f est une P-fonction sur \mathbb{R}^m alors, pour chaque orthant \mathbb{O} inclus dans \mathbb{R}^m , il existe au moins un point complémentaire de f , Λ appartenant à l'orthant.

Démonstration : Prenons dans l'orthant \mathbb{O} deux points complémentaires distincts Λ et Γ

$$\Lambda = x + f(x)$$

$$\Gamma = y + f(y)$$

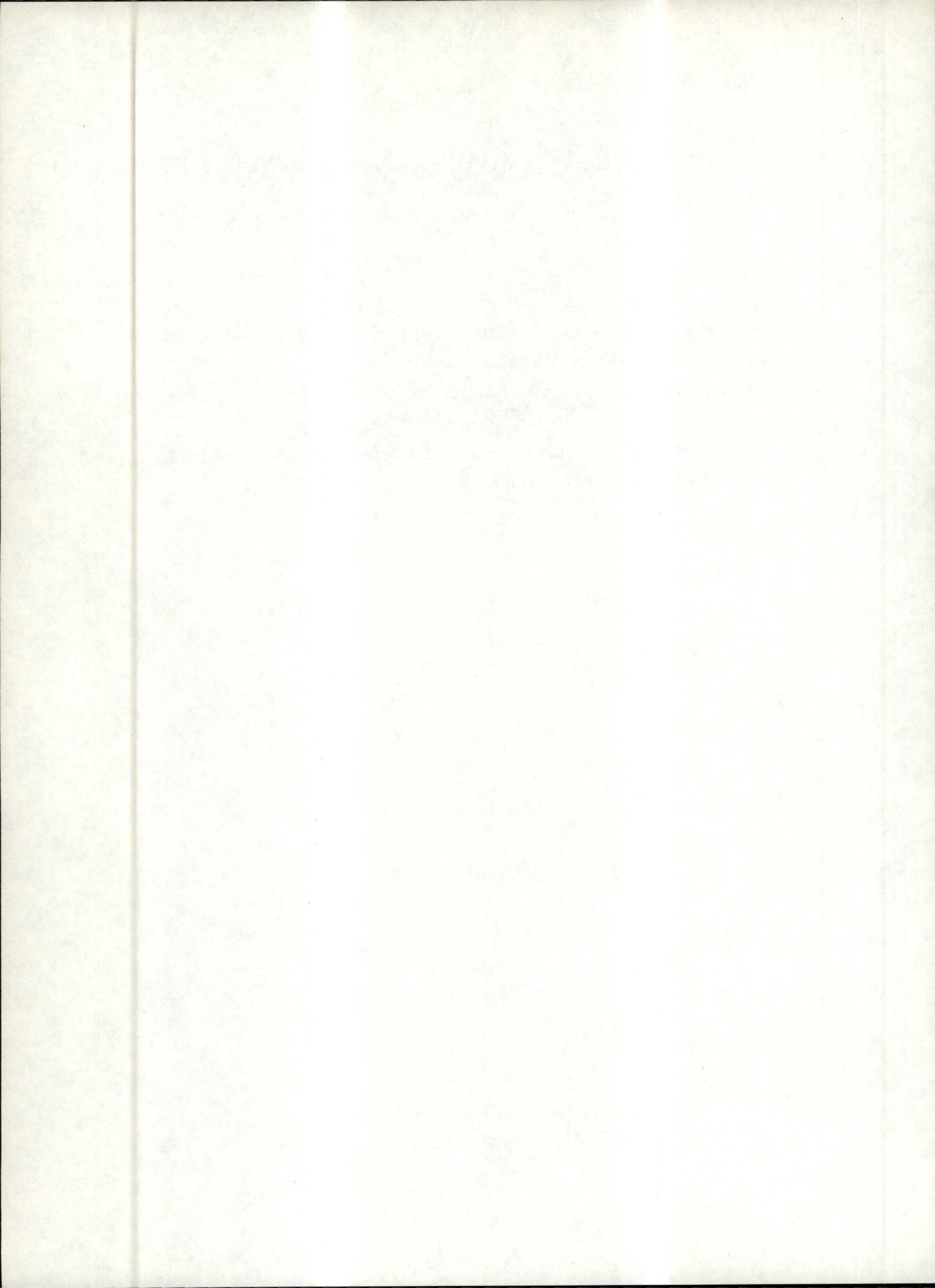
Pour chaque indice i de N , vu la condition de complémentarité (3), on a :

$$\begin{aligned} (\alpha_i - \gamma_i)(f_i(x) - f_i(y)) &= \alpha_i f_i(x) - \alpha_i f_i(y) - \gamma_i f_i(x) + \gamma_i f_i(y) \\ &= 0 - \alpha_i f_i(y) - \gamma_i f_i(x) + 0 \\ &\leq 0 \end{aligned}$$

La dernière inégalité provient du fait que les points s et t appartiennent au même orthant \oplus

Cette même dernière inégalité contredit l'hypothèse : f est une P-fonction sur \mathbb{R}^n tout entier.

Ainsi donc, il existe au moins un point complémentaire de f appartenant à l'orthant \oplus .



Nous venons donc de montrer que si l'application f est PBJM, il existe 2^n points complémentaires (2.3) et au moins un point complémentaire par orthant (2.4). Dès lors, comme il y a 2^n orthants dans l'espace R^n , le principe du "trou du pigeon", précédemment énoncé, exige qu'il y ait un et un seul point complémentaire par orthant.

En effet, dans notre cas, considérons les ensembles \mathcal{X} et \oplus

$$g: \mathcal{X} \rightarrow \oplus$$

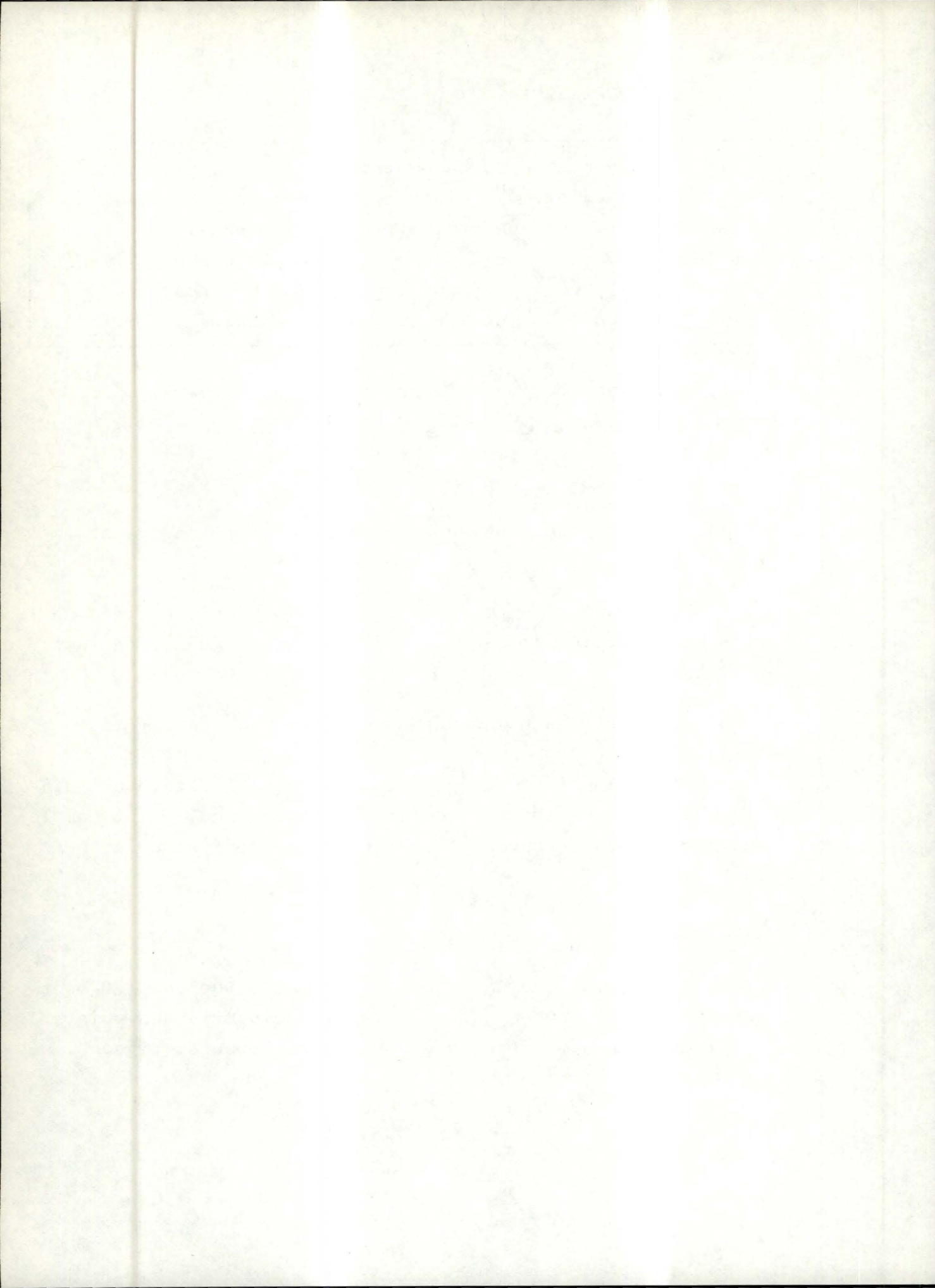
Nous venons de voir que

- 1) Il existe au plus une solution par orthant (g est fonctionnelle) ;
- 2) Toute solution possède une certaine configuration de signes, (correspondant à un certain orthant) (g est partout défini);
- 3) Il existe au moins une solution par orthant (g est surjective).

Dès lors, l'application g considérée est bijective ce qui signifie: chaque orthant contient un et un seul point complémentaire. Nous pouvons encore exprimer ceci sous la forme suivante :

Théorème 2.5.

Supposons que la fonction f considérée soit PBJM et que le cpf soit non dégénéré; alors, l'ensemble des points complémentaires vérifie la propriété du "trou du pigeon".



Résultat fondamental

Si les applications S et g sont PBJM, l'application $g \circ S : \mathcal{P}(N) \rightarrow \mathbb{H}$ est une bijection.

Ce résultat est fondamental car c'est une des conditions sous-lesquelles les algorithmes directs présentés au chapitre suivant possèdent une solution unique.

6

LA PROPRIETE DU "TROU DU PIGEON" POUR LES P-FONCTIONS UNIFORMES

=====

La propriété du "trou du pigeon" est vérifiée, nous venons de le montrer, pour une fonction dont la matrice jacobienne est bornée positivement. Nous allons montrer qu'elle est également vérifiée pour une sous-classe des P-fonctions : les P-fonctions uniformes.

Définition 2.9

On appelle cône rectangulaire tout cône du type :

$$\prod_{i=1}^m A_i \quad \text{où } A_i \text{ est soit } - \begin{cases} \mathbb{R}_+ \\ \mathbb{R}_- \\ \mathbb{R} \end{cases}$$

Définition 2.10

La fonction $f : \mathbb{R}^m \longrightarrow \mathbb{R}^m$ est une P-fonction uniforme sur un ensemble T si pour chaque paire d'éléments distincts x, y de T, on a :

$$\max_{i \in N} (x_i - y_i) (f_i(x) - f_i(y)) \geq c \|x - y\|^2 \quad \forall c > 0$$

Théorème 2.6

Supposons que la fonction f de \mathbb{R}^m dans \mathbb{R}^m soit continue sur le cône rectangulaire K_r et que cette fonction f soit une P-fonction uniforme sur K_r ;

alors, il existe un unique vecteur x de K_r , avec $f(x)$ appartenant à K_r^* (le cône conjugué), tel que $x^T \cdot f(x) = 0$

Démonstration : La démonstration est établie [17] pour un problème de complémentarité non linéaire généralisé que l'on définit comme suit :

Soient f une fonction continue de l'espace \mathbb{R}^m dans lui-même K un cône et K^* son cône conjugué.

Trouver, s'il existe, un vecteur x de \mathbb{R}^m vérifiant les propriétés suivantes :

$$1) \forall x \in K \quad f(x) \in K^* = \{y \mid y^T x \geq 0 \quad \forall x \in K\}$$

$$2) \forall x \in K \quad x^T f(x) = 0$$

Remarque

Chaque orthant \oplus de \mathbb{R}^m est un cône rectangulaire dont le cône conjugué est \ominus lui-même.

Théorème 2.7

Supposons que le cpf considéré soit non-dégénéré.

Si $f(x)$ est une P-fonction sur l'ensemble \mathbb{R}^m , alors l'ensemble des points complémentaires, \mathcal{X} , vérifie la propriété du "trou du pigeon".

Démonstration : On applique le théorème 2.6. à chaque orthant de \mathbb{R}^m ;

Contre-exemple : Cet exemple nous montre l'importance de l'hypothèse : f est une P-fonction uniforme.

Soit $f: \mathbb{R} \longrightarrow \mathbb{R}$
 $x \longmapsto f(x) = -e^{-x} - x$

Cette fonction est différentielle et sa dérivée est strictement positive pour tout élément x de \mathbb{R} .

D'autre part, comme $f(x)$ est strictement négative pour tout élément x de \mathbb{R} , l'ensemble des points complémentaires ne véri-

ne vérifie pas la propriété du "trou du pigeon". Le seul point complémentaire provient de $x = 0$.

La fonction n'est effectivement pas une P-fonction uniforme. Cependant, c'est une P-fonction sur \mathbb{R} . En effet, par le théorème de la valeur moyenne, nous avons

$$(x - y) (f(x) - f(y)) > 0$$

pour tout $x \neq y$ de \mathbb{R}

7 LA PROPRIÉTÉ DU "TROU DU PIGEON" DANS LE CAS LINÉAIRE

Le cas linéaire étant un cas particulier du cas non linéaire, nous adapterons d'abord le théorème 2.5 et 2.7 aux fonctions affines. Nous montrerons ensuite que, dans le cas d'un LCP, la propriété de "trou du pigeon" est une caractéristique des P-matrices.

A Lien avec les P-matrices

Théorème 2.8

Soit A une matrice $n \times n$. Les deux assertions suivantes sont équivalentes : 1) Tous les mineurs principaux de A sont positifs;

2) Si pour chaque vecteur x non nul, on a $y = Ax$ alors, il existe un indice i tel que $x_i y_i > 0$

Pour la démonstration : voir [18].

Théorème 2.9

Supposons que la fonction f de \mathbb{R}^m dans \mathbb{R}^m soit une fonction affine, c'est-à-dire $f(x) = Mx + q$

Si f appartient à PBJM ou si f est une P-fonction uniforme sur \mathbb{R}^m , alors la matrice M est une P-matrice.

Démonstration : 1) Considérons f appartenant à PBJM.

Trivialement, comme f possède une matrice jacobienne M constante, tous ses mineurs principaux sont strictement positifs puisque bornés par δ et δ^{-1} , le nombre δ étant strictement positif. Donc la matrice M est une P-matrice.

2) Supposons que f soit une P-matrice uniforme sur \mathbb{R}^m

$$\mu = x - y \text{ et } r = M\mu$$

Si $\mu \neq 0$ alors, par définition, d'une P-fonction uniforme, $\max_{i \in N} \mu_i r_i \geq c \|\mu\|^2 > 0$

Donc, par le théorème 2.8; M est une P-matrice.

B Caractérisation des ensembles A_M et H_M

Ces ensembles A_M et H_M nous permettront, dans le paragraphe suivant, de vérifier la propriété du "trou du pigeon" pour des P-matrices.

Définition 2.11.

$A_M = \{q \in \mathbb{R}^m \mid q \text{ est non-dégénéré par rapport à } M \text{ (définition : voir lexique)}\}$.

$H_M = \{q \in A_M \mid \text{Le LCP possède des points complémentaires qui vérifient la propriété du "trou du pigeon"}\}$.

Théorème 2.10.

Si M est une P-matrice, alors A_M est égal à H_M .

Démonstration : Par définition, on a $H_M \subseteq A_M$

De plus, comme M a ses mineurs principaux positifs et constants, par le théorème 2.7., on a $A_M \subseteq H_M$

Donc, $A_M = H_M$ si M est une P-matrice.

Lemme 2.3.

Si $q \in A_M$ alors, la cardinalité de l'ensemble \mathcal{X} est inférieure ou égale à 2^n . On obtient l'égalité lorsque M est une matrice non-dégénérée (définition : voir lexique).

Démonstration : L'ensemble \mathcal{X} des points complémentaires est l'ensemble des solutions du système $B^{(k)} \Delta^{(k)} = q$ où $k = 1, 2, \dots, 2^n$ où k dénombre les points complémentaires et les matrices de base complémentaire correspondantes.

Chaque colonne de $B^{(k)}$ est soit $I_{.j}$ si $\Delta_j^{(k)} = y_j$
soit $-M_{.j}$ si $\Delta_j^{(k)} = x_j$

D'autre part, si chaque $[B^{(k)}]^{-1}$ existe, la matrice M est non-dégénérée. Il existe donc exactement une solution $\Delta^{(k)}$ pour $k=1, 2, \dots, 2^n$. Cependant, s'il existe un indice k_n pour lequel $B^{(k_n)}$ est singulière, la matrice M est une matrice dégénérée. Deux possibilités peuvent alors se présenter.

Ou bien il n'existe pas de solution, ou bien il en existe une infinité. Cette dernière possibilité est trivialement en contradiction avec le lemme 2.1. Dans le cas où la résolution du système $B^{(k_n)} \Delta^{(k_n)} = q$ ne fournit aucune solution, la cardinalité de l'ensemble \mathcal{X} est évidemment inférieure ou égale à 2^n .

Théorème 2.11

Si M est une matrice non-dégénérée, alors le nombre de solutions d'un LCP linéaire à la même parité pour tout

q de A_M .

Pour la démonstration : voir [19].

Théorème 2.12

Si M est une matrice non-dégénérée, alors le nombre de solutions d'un GCP linéaire a la même parité pour tout q de A_M . On définit un GCP linéaire par le problème suivant :

Considérons l'orthant $\Theta \subseteq \mathbb{R}^m$

Trouver le vecteur x de \mathbb{R}^m tel que, si $y = Mx + q$ alors, x appartient à Θ , y appartient à $\Theta^* = \Theta$ et $y^T x = 0$.

Démonstration : Considérons l'homéomorphisme dans \mathbb{R}^m qui envoie, en particulier, l'orthant Θ sur \mathbb{R}_+^m et tel que,

$\lambda(x) = Dx$ où D est une matrice diagonale;

$$D = \text{diag}(d_1, d_2, \dots, d_n)$$

où $d_i = \text{sgn}(x_i)$ pour tout x appartenant à Θ

On a donc : $D = D^{-1} = D^T$

Supposons que $X = Dx$ et $y = Dy$

Ainsi, $y^T X = (Dy)^T Dx = y^T D^T D x = y^T x$

Soit $M = DM D^{-1} = DMD$

On a donc que M est une matrice non-dégénérée si M est non-dégénérée.

Soit \hat{q} est un élément de A_M et $Q = D\hat{q}$.

Considérons d'autre part, le LCP ordinaire rapporté à (Q/M) .

Supposons que $Q = y - Mx$ et qu'il y a au maximum $(n-1)$ variables de (y, x) non nulles.

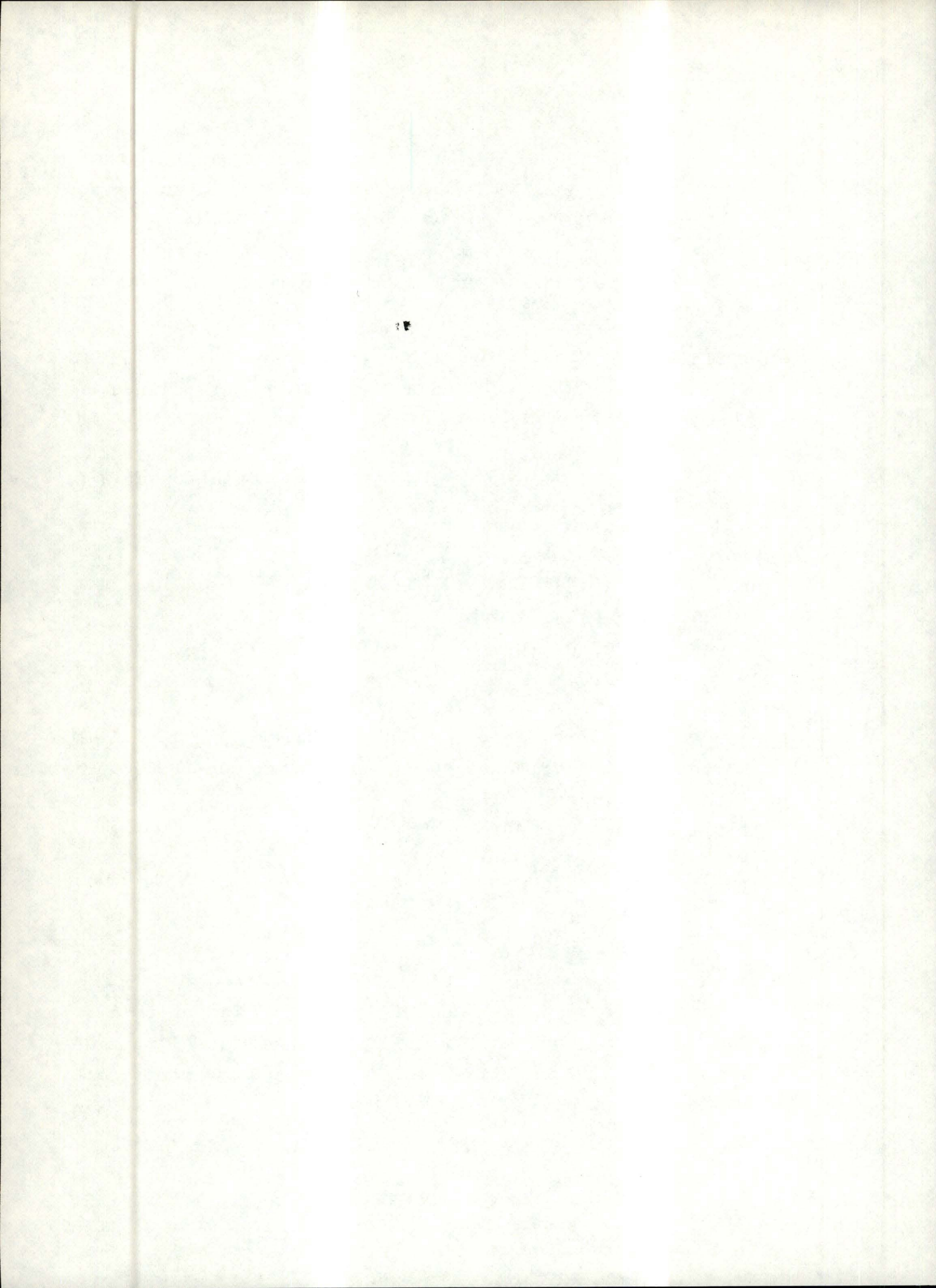
Multiplions $Q = y - Mx$ par D^{-1}

On a $D^{-1} Q = D^{-1} y - D^{-1} Mx$

$$\hat{q} = y - D^{-1} (DMD) Dx$$

$$= y - Mx$$

On a donc toujours au maximum $(n-1)$, des $2n$ variables, non nulles : ce qui contredit le fait que \hat{q} est un élément de A_M . C'est pourquoi, il est impossible d'exprimer Q comme combi-



raison linéaire d'au maximum $(n-1)$ colonnes de l'ensemble

$$\{I_{.1}, \dots, I_{.m}, -m_{.1}, \dots, -m_{.m}\}$$

en ayant, en même temps q appartenant à A_m .

De plus, x appartient à \mathbb{H} si, et seulement si, on a :

$$x \geq 0, y \in \mathbb{H}^* = \mathbb{H}$$

c'est-à-dire si, et seulement si :

$$y \geq 0 \text{ et } x^T y = x^T y$$

On peut donc dire que les solutions du GCP linéaire sur \mathbb{H} sont en bijection avec les solutions du LCP rapporté à $h(x)$. Par le théorème 2.11, et en appliquant au LCP l'application réciproque $h^{-1}(x)$, la thèse est démontrée.

Théorème 2.13.

Si le nombre de solutions d'un LCP est constant pour tout q appartenant à A_M , alors cette constante est égale à l'unité et l'ensemble des cônes complémentaires d'intérieur non vide forme une partition de \mathbb{R}^m .

Pour la démonstration : voir [19].

Théorème 2.14.

Soit M une matrice non-dégénérée. L'ensemble des cônes complémentaires forme une partition de \mathbb{R}^m si, et seulement si, la matrice M est une P-matrice.

Pour la démonstration : voir [22].

Théorème 2.15.

Si M n'est pas une P-matrice, alors l'ensemble H_M est l'ensemble vide.

Démonstration : (Par contraposition).

Supposons que H_M est non vide et que q est un de ses éléments.

Alors, par le lemme 2.3., la matrice M est non-dégénérée.

Supposons ensuite qu'il existe un élément q^* de A_M qui n'appartient pas à H_M . Il existe donc un orthant \oplus inclus dans \mathbb{R}^m , tel que χ_{q^*} ait une intersection vide avec \oplus .

χ_{q^*} est l'ensemble des points complémentaires du LCP rapporté à (q^*/M) .

Ce fait contredit le théorème 2.12 puisque le nombre de solutions du GCP sur \oplus n'est pas de même parité.

On peut donc dire que H_M est égal à A_M et qu'il y a un nombre constant de solutions au LCP pour tout q appartenant à A_M .

Les théorèmes 2.13 et 2.14 nous permettent alors de conclure que M est une P-matrice.

C Une caractérisation des P-matrices : la propriété du "trou du pigeon"

Le théorème 2.9 nous certifie que si $f \in \text{PBIM}$ ou f est une P-fonction uniforme, alors M est une P-matrice.

D'autre part, le théorème 2.16 ci-dessous affirme que si M est une P-matrice, alors le LCP associé à M possède des points complémentaires qui vérifient la propriété du "trou du pigeon". Ainsi donc, nous avons établi, pour le cas linéaire, la condition sous laquelle une fonction affine vérifie la propriété du "trou du pigeon".

Théorème 2.16.

Pour une matrice M , $n \times n$, les assertions suivantes sont équivalentes :

$$1) M \in (P)$$

C'est-à-dire M est une P -matrice;

$$2) H_M \neq \emptyset$$

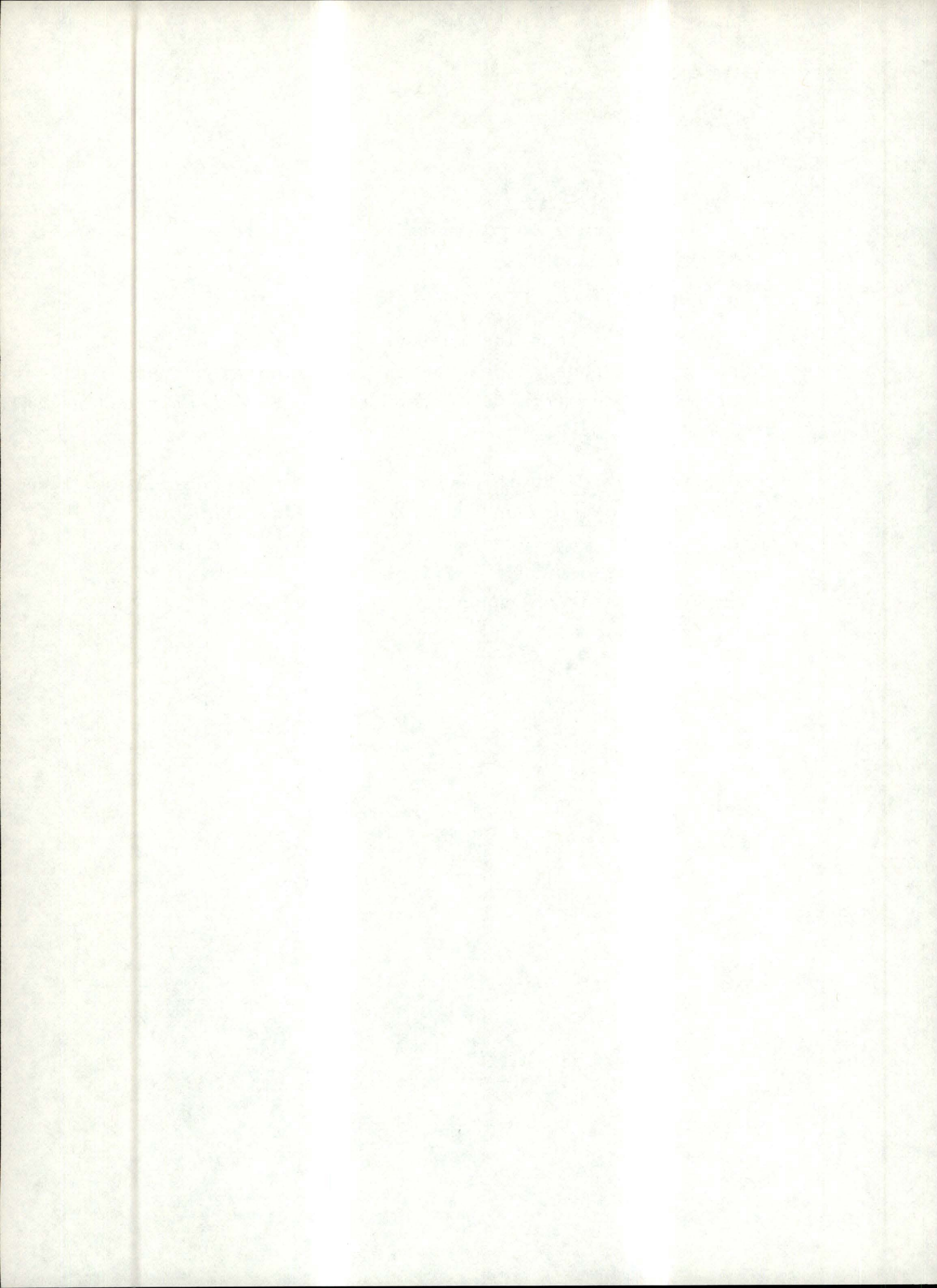
C'est-à-dire $\exists q \in A_M$ tel que le LCP correspondant possède des points complémentaires qui vérifient la propriété du "trou du pigeon";

$$3) H_M = A_M$$

C'est-à-dire tous les vecteurs q pour lesquels le LCP correspondant possède des points complémentaires qui vérifient la propriété du "trou du pigeon" sont non-dégénérés par rapport à la matrice M .

Démonstration : Le théorème 2.10 démontre que 1) implique 3). A_M est toujours un ensemble non vide. En effet, son complémentaire est fermé et de dimension égale au plus à $(n-1)$. C'est pourquoi 3) implique 2).

Et en prenant la contraposition du théorème 2.15, on sait que 2) implique 1).



C H A P I T R E I I I

LES ALGORITHMES DIRECTS

-----I N T R O D U C T I O N

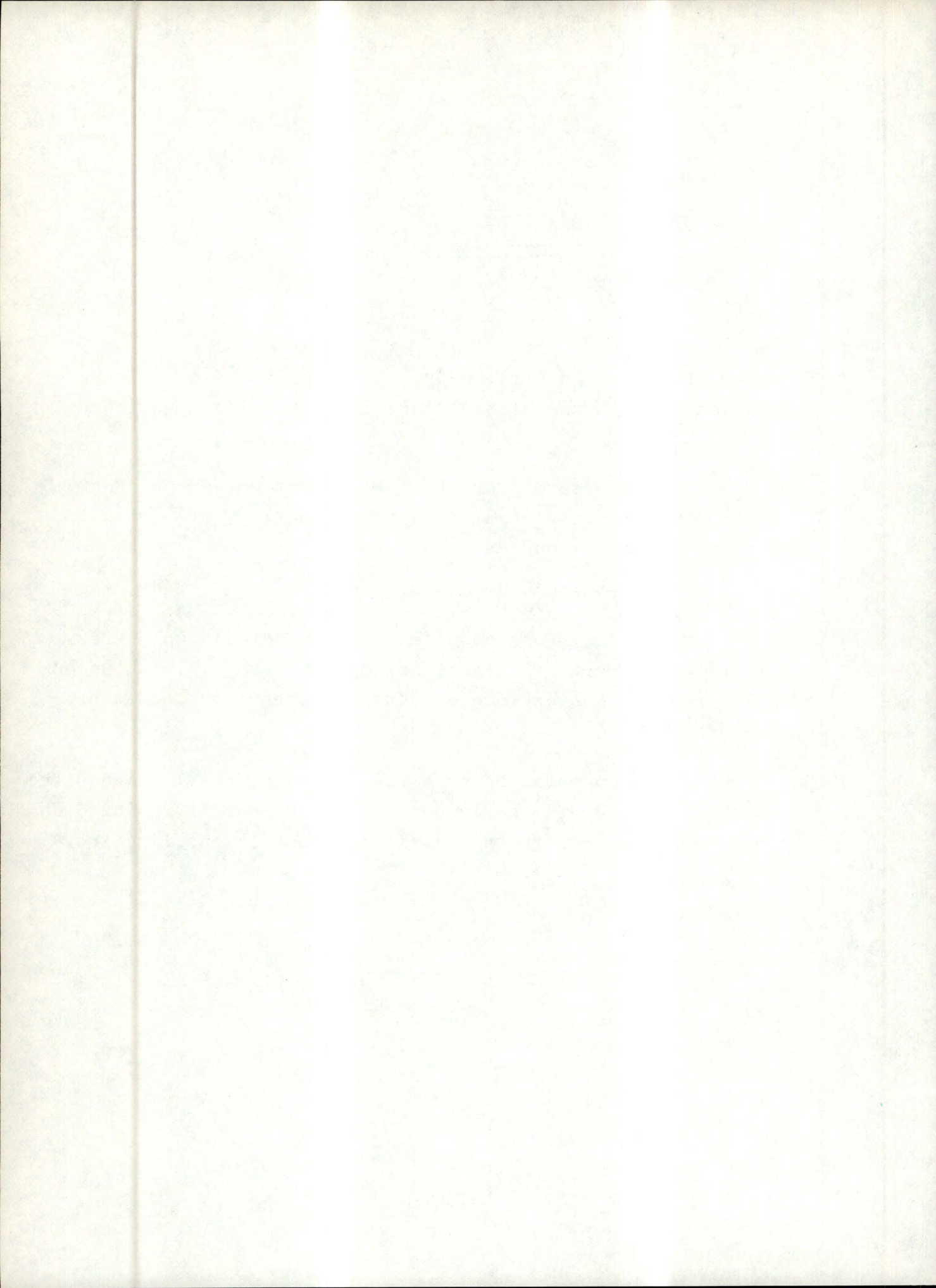
Dans ce chapitre, nous allons définir les notions de base et décrire une classe d'algorithmes : les algorithmes directs.

Nous exposerons brièvement deux exemples typiques d'algorithmes appartenant à cette classe :

- l'algorithme de Bard
- l'algorithme de Murty.

Nous étudierons ensuite, plus en détail, un autre algorithme direct : l'algorithme de Kostreva [12]. Nous en montrerons la convergence et nous le comparerons aux deux précédents.

Cette comparaison sera illustrée en détail au chapitre IV où l'on trouvera une étude statistique sur la rapidité de convergence de ces trois algorithmes dans le cas linéaire.



1 LES ALGORITHMES DIRECTS

A Définitions

Rappelons que le problème de complémentarité relatif à la fonction f (notation : cpf) est le suivant :

Soit $f : \mathbb{R}^m \longrightarrow \mathbb{R}^m$ une fonction continue.

Trouver, s'il existe, un élément x de \mathbb{R}^m qui vérifie les conditions :

$$\forall i \in N = \{1, 2, \dots, m\} \quad 1) \quad x_i \geq 0$$

$$2) \quad f_i(x) \geq 0$$

$$3) \quad x_i f_i(x) = 0$$

Un algorithme direct pour cpf est un algorithme qui génère une suite finie $\{I^{(k)}\}_{k=1}^m$ d'ensembles d'indexations de l'ensemble de parties de $N = \{1, 2, \dots, m\}$ afin de trouver l'unique solution positive du cpf.

Remarque : le cpf défini ci-dessus, ne possède pas toujours une solution sous la seule hypothèse : f est une fonction continue.

Nous nous limiterons, dans ce présent travail, au cpf qui possède une solution unique, c'est-à-dire (comme nous venons de le montrer au chapitre précédent) que si f est PBJM, alors, il existe un, et un seul, point complémentaire par orthant. Nous ne considérerons que les cpf non dégénérés associés à une fonction dont la matrice jacobienne est bornée positivement.

C'est pourquoi, si x^* satisfait les conditions 1), 2), 3) alors il existe au moins un ensemble d'indexation $I^* \subseteq N$ tel que :

$$\begin{cases} f_i(x^*) = 0 \\ x_i^* = 0 \end{cases}$$

$$\forall i \in I^*$$

$$\forall i \in N \setminus I^*$$

$$\text{et } \left\{ \begin{array}{l} f_i(x^*) > 0 \\ x_i^* > 0 \end{array} \right. \quad \begin{array}{l} \forall i \in N \setminus I^* \\ \forall i \in I^* \end{array}$$

autrement dit $\lambda^* = x^* + f(x^*) > 0$

Cependant, d'après les résultats rappelés ci-dessus, nous pouvons également résoudre, et de la même manière, le problème suivant :

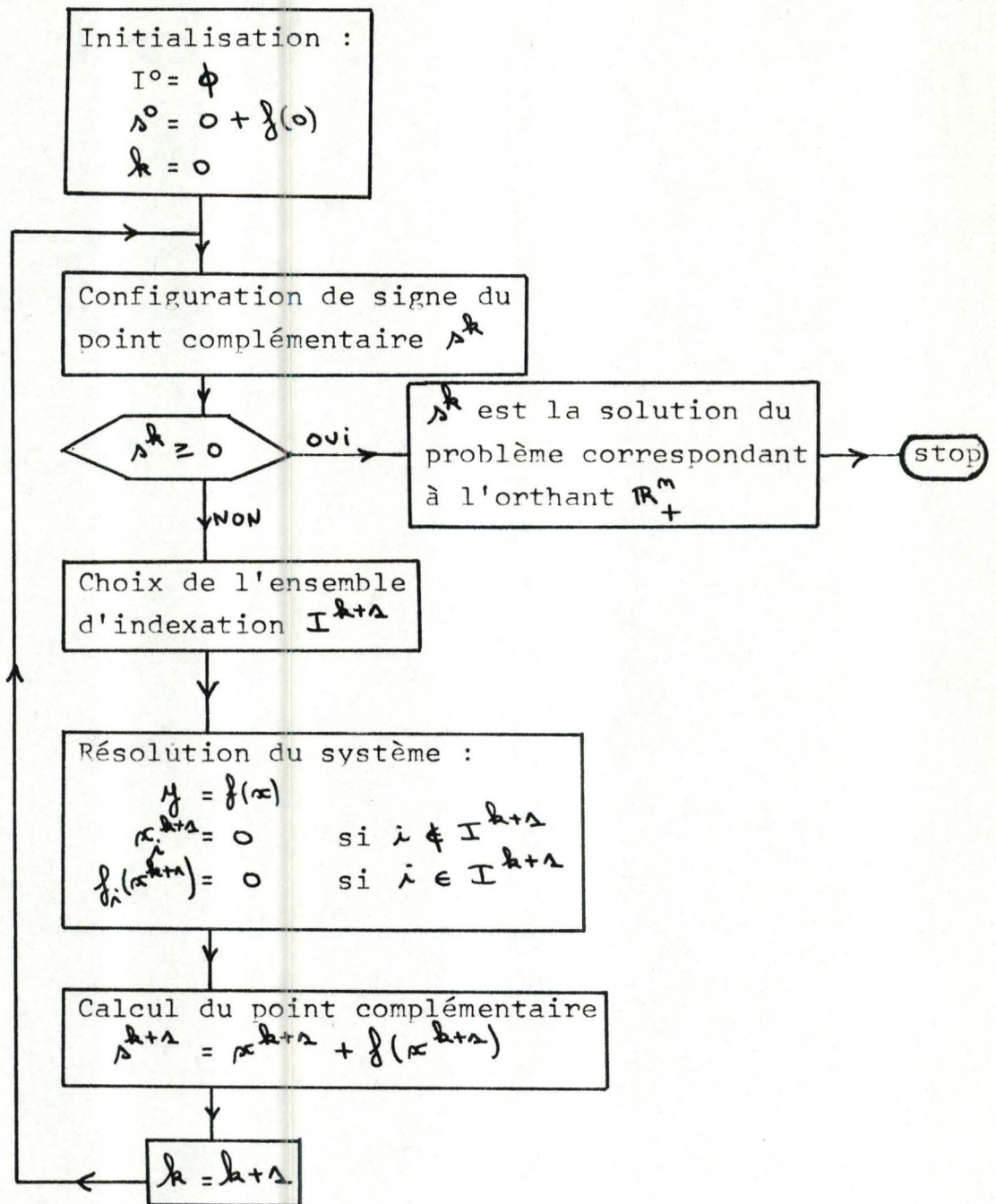
Soit $f : \mathbb{R}^m \longrightarrow \mathbb{R}^m$ une fonction PBJM.

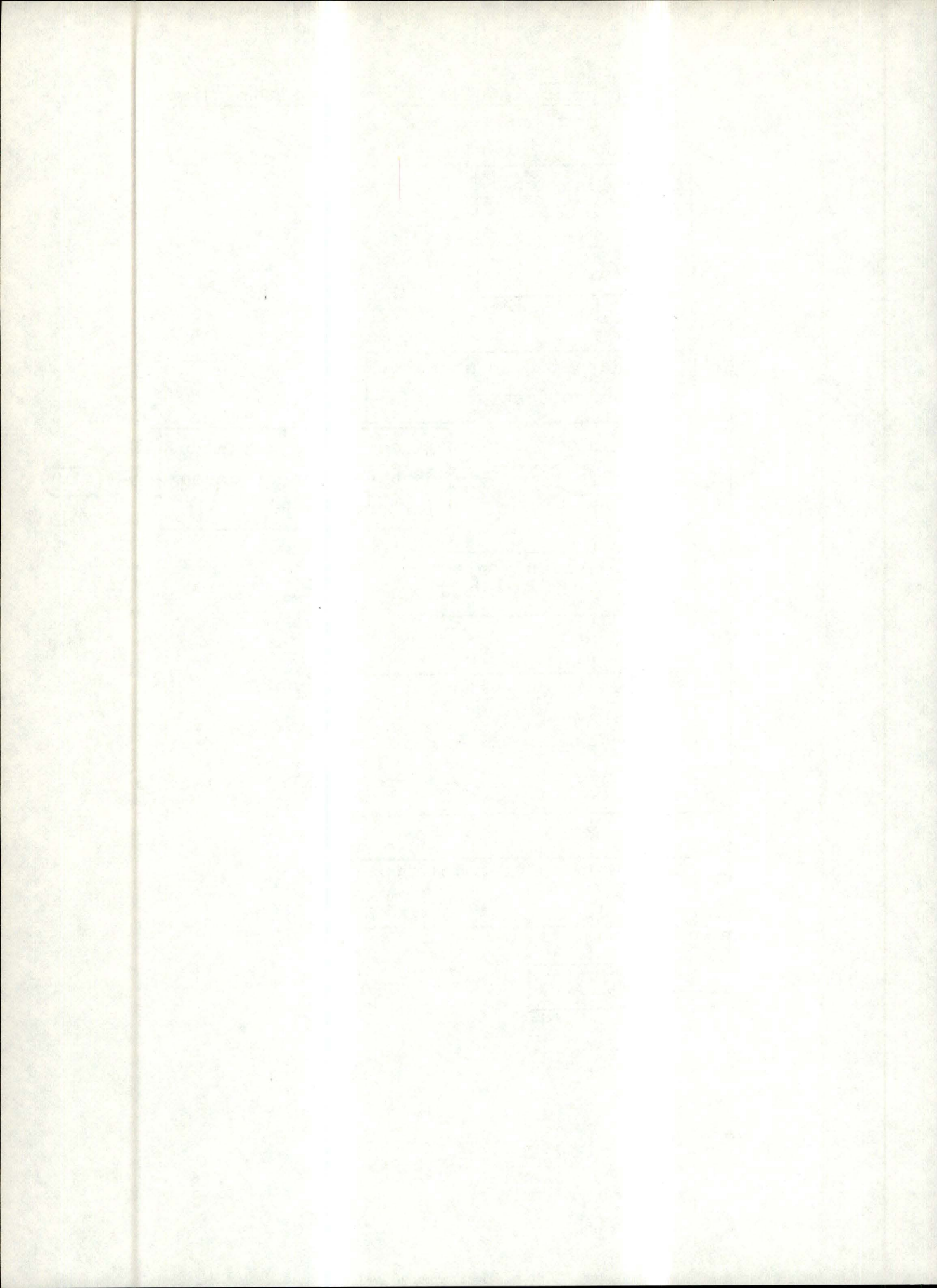
Trouver, s'il existe, le vecteur x de \mathbb{R}^m qui vérifie les conditions :

$$1) x_i f_i(x) = 0 \quad \forall i \in N$$

2) la configuration de signe du point complémentaire

$$\lambda = x + f(x) \quad \text{étant fixée.}$$

B Organigramme général d'un algorithme direct



C Remarque

Cet algorithme porte un intérêt particulier à la configuration de signe du point complémentaire. Dès lors, pour être suffisamment précise, la solution des systèmes

$$\left\{ \begin{array}{l} x_i^k = 0 \\ f_i(x^k) = 0 \end{array} \right. \quad \begin{array}{l} \forall i \notin I^k \\ \forall i \in I^k \end{array}$$

devra posséder la même configuration de signes que celle de l'orthant imposé.

Dans cet algorithme, on chemine de solution complémentaire en solution complémentaire.

D Les différents types d'algorithmes directs

1) Un algorithme direct qui génère une suite d'ensembles d'indexation, où $I^{(k+1)}$ diffère de $I^{(k)}$ par la propriété mathématique

$$I^{(k+1)} = (I^{(k)} \setminus \{i_d\}) \cup \{i_a\} \quad \forall k$$

où i_a est un indice qui s'ajoute dans l'ensemble d'indexation, et

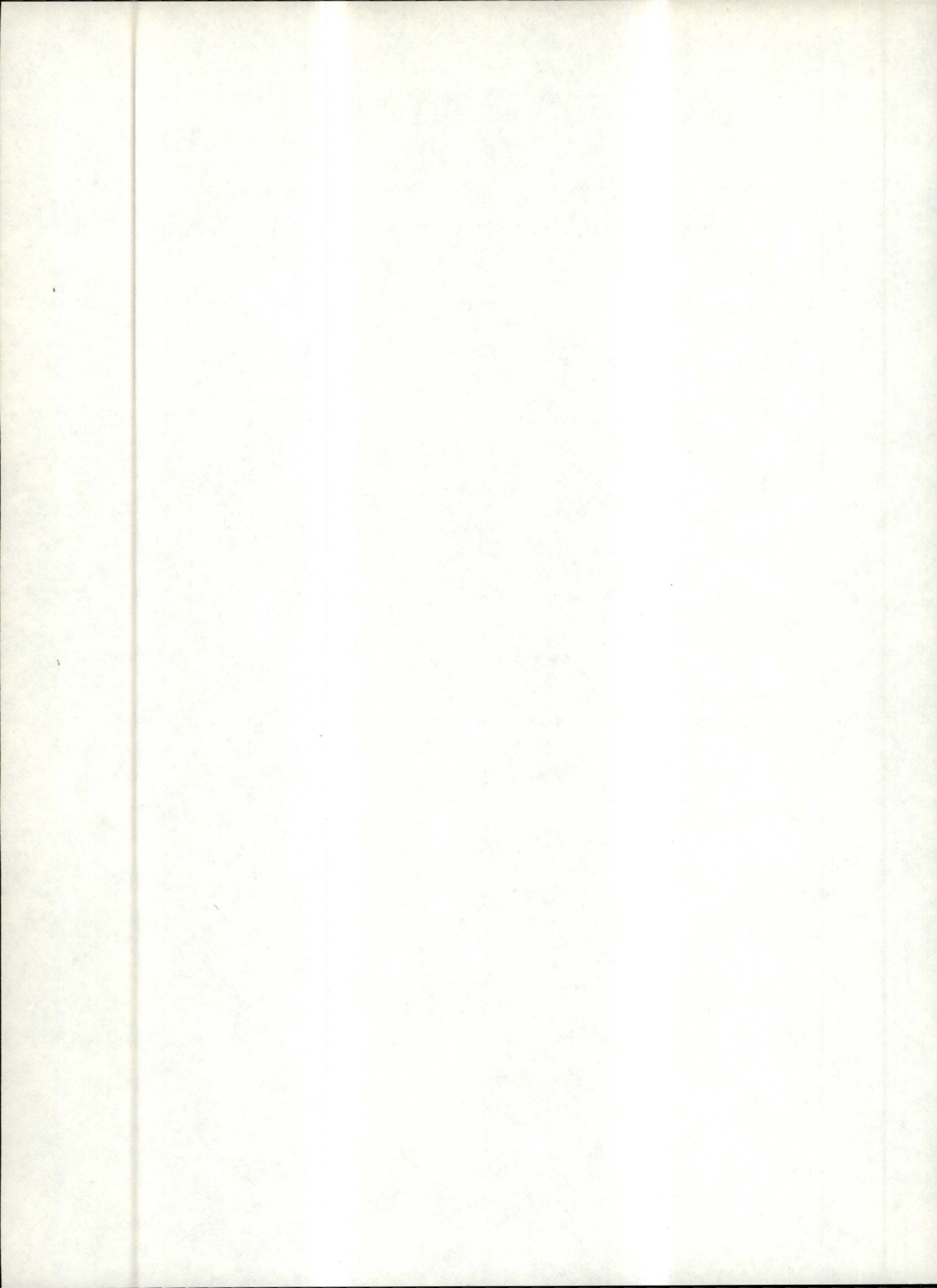
où i_d est un indice qui se retire de l'ensemble d'indexation, est appelé "algorithme direct de pivot unique".

2) Un algorithme direct qui génère une suite d'ensemble d'indexation, où $I^{(k+1)}$ diffère de I^k par plus d'un indice, est appelé "algorithme direct de pivot bloc".

Dans ce cas, la propriété mathématique devient :

$$I^{(k+s)} = \{ I^{(k)} \setminus I_d^{(k)} \} \cup I_a^{(k)}$$

où $I_d^{(k)}$ indique l'ensemble des indices qui quittent $I^{(k)}$
 et $I_a^{(k)}$ indique l'ensemble des indices qui s'ajoutent à $I^{(k)}$



2

DEUX EXEMPLES D'ALGORITHMES DIRECTS DE PIVOT UNIQUE

A Algorithme de Bard (20)

Description :

Pas 1 : Initialisation : $I^0 = \phi$
 $x^0 = 0$
 $s^0 = 0 + f(0)$
 $k = 0$

aller au pas 2.

Pas 2 : Les composantes du point complémentaire s^k sont-elles toutes non-négatives ?
 Si oui, ce point complémentaire s^k est la solution du problème.
 Sa configuration de signes correspond à l'orthant \mathbb{R}_+^m .
 Si non, aller au pas 3.

Pas 3 : Recherche de la plus petite composante négative du point complémentaire, c'est-à-dire :

$$\min_{j < 0} \{s_j^k\}_{j \in N} = s_i^k$$

Aller au pas 4.

Pas 4 : s_i^k est-elle une composante du vecteur x ?
 Si oui, l'ensemble d'indexation suivant est :

$$I_{k+1} = I_k \setminus \{i\}$$

Si non, l'ensemble d'indexation suivant est :

$$I_{k+1} = I_k \cup \{i\}$$

Aller au pas 5.

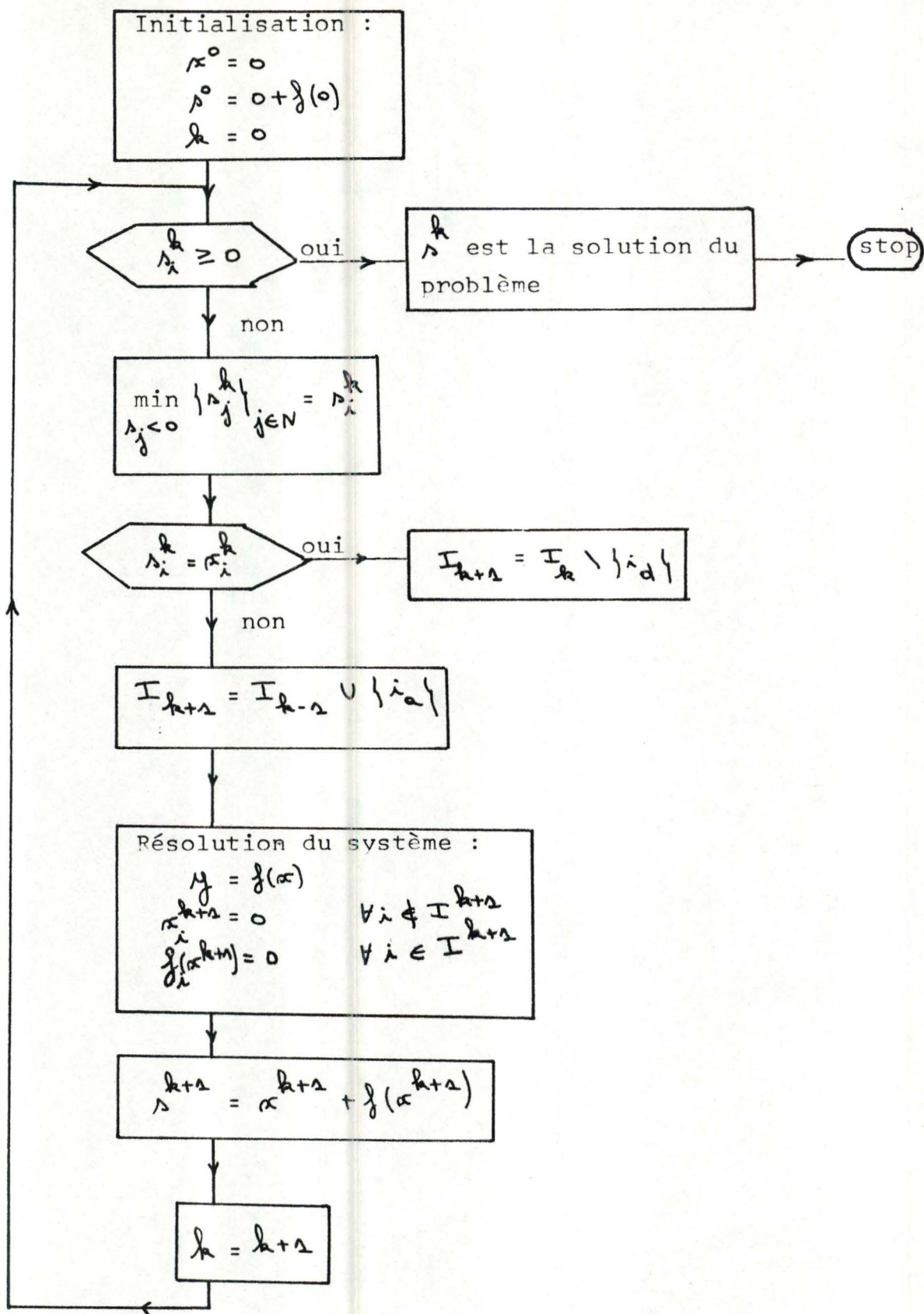
Pas 5 : Résolution du système :

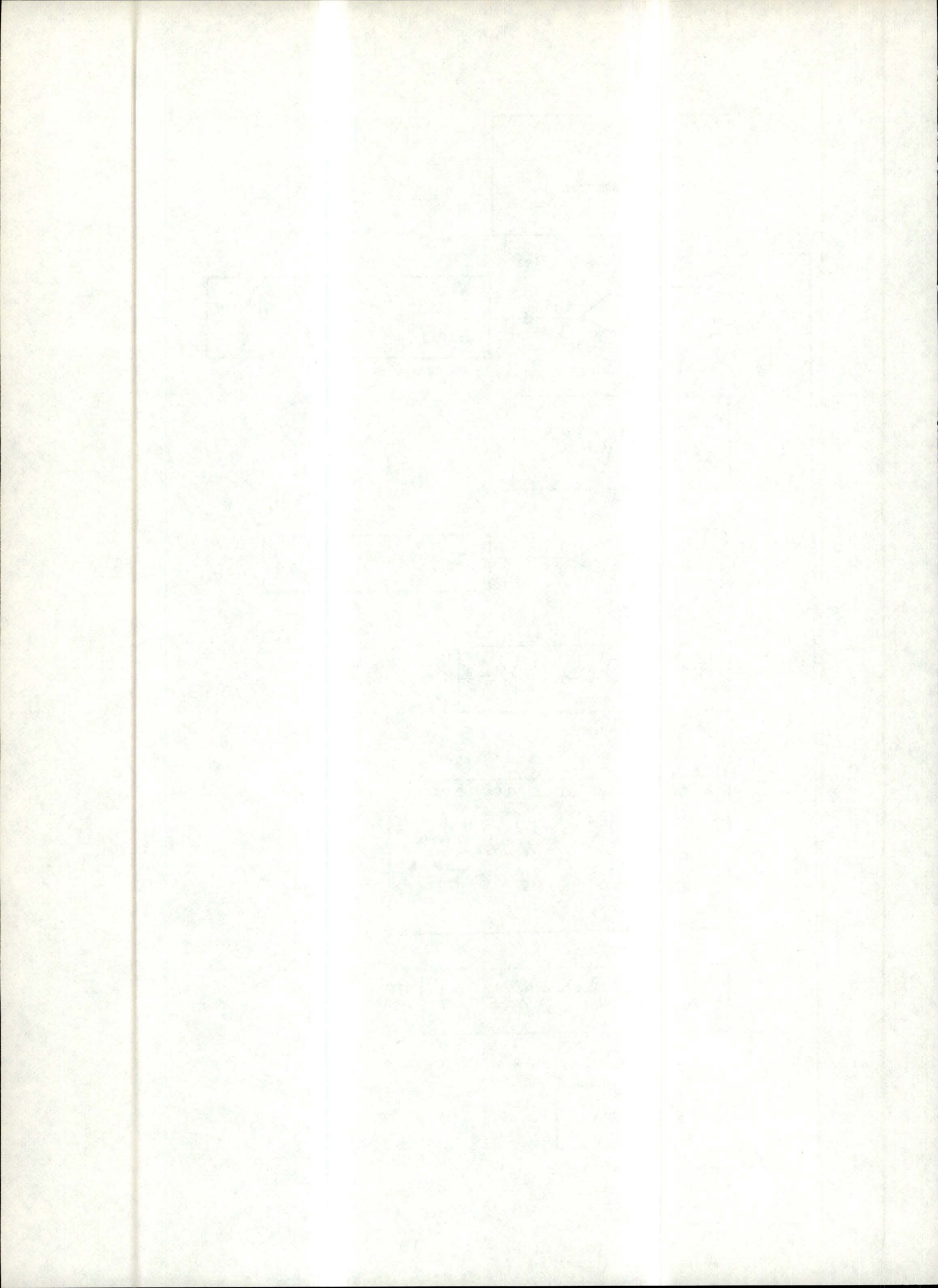
$$\left\{ \begin{array}{l} y = f(x) \\ x_i^{k+1} = 0 \quad \text{si } i \notin I^{k+1} \\ f_i(x^{k+1}) = 0 \quad \text{si } i \in I^{k+1} \\ \text{Aller au pas 6} \end{array} \right.$$

Pas 6 : Calcul du point complémentaire $\lambda^{k+1} = x^{k+1} + f(x^{k+1})$
Aller au pas 7.

Pas 7 : On vient d'accomplir une itération complète.
On passe à l'itération suivante : $k = k+1$
Aller au pas 2.

Organigramme





Remarques : 1) On observe facilement que l'algorithme de Bard est un algorithme direct de type pivot unique. Le pivot est choisi comme l'élément correspondant à la composante la plus négative du point complémentaire.

2) A notre connaissance, la démonstration de la convergence de cet algorithme n'est toujours pas acquise.

Exemple numérique : Dans cet exemple, nous poserons $y = f(x)$ avec : $f: \mathbb{R}^2 \longrightarrow \mathbb{R}^2$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightsquigarrow \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} -3 + 2x_1 - x_2 \\ -2 + 3x_1 + x_2 \end{pmatrix}$$

1) $I_0 = \emptyset$

Dès lors, le système à résoudre est :

$$\begin{cases} x_1 = 0 \\ x_2 = 0 \\ y_1 = -3 + 2x_1 - x_2 \\ y_2 = -2 + 3x_1 + x_2 \end{cases}$$

La solution immédiate est : $\lambda_0 = (-3, -2)$

L'ensemble d'indexation suivant est donc : $I_\lambda = \emptyset \cup \{1\} = \{1\}$

2) $I_\lambda = \{1\}$

Dès lors, le système à résoudre est :

$$\begin{cases} y_1 = 0 \\ x_2 = 0 \\ y_1 = -3 + 2x_1 - x_2 \\ y_2 = -2 + 3x_1 + x_2 \end{cases}$$

Ce qui conduit à un échange des variables x_2 et y_2 et donc, à un pivotage sur l'élément 2.

	1	x_1	x_2
y_1	-3	2	-1
y_2	-2	3	1

	1	y_1	x_2
x_1	3/2	1	1/2
y_1	5/2	3/2	5/2

La solution est : $x_1 = (3/2, 5/2)$

Elle est positive : donc la solution du problème est :

$$(x_1, x_2, y_1, y_2) = (3/2, 0, 0, 5/2)$$

Exemple : programmé de dimension 5 dans le cas linéaire :

Le programme se trouve au chapitre IV du mémoire.

Il faut cependant remarquer que la programmation est réalisée pour une dimension maximale de 10 et que

- le point complémentaire est noté q ,
- le vecteur x est noté z ,
- le vecteur $y = f(x)$ est noté w .

La matrice de données est une matrice générée aléatoirement. Elle est ensuite transformée en une matrice à diagonale fortement dominante afin que le problème considéré possède une solution unique, la matrice étant une P-matrice.

METHODE DE BARD

LE TABLEAU DE L'ITERATION D EST :

	Z01	Z02	Z03	Z04	Z05	Q
W01	34.20	-6.51	5.69	-7.25	5.33	-2.82
W02	-4.86	19.27	0.87	-0.48	9.25	-0.15
W03	-4.17	-3.66	20.37	6.43	-1.70	-8.05
W04	7.00	-5.48	4.05	28.38	5.50	-9.91
W05	-9.00	-4.79	-7.73	-3.28	34.69	8.85

LE POINT DE COMPLEMENTARITE EST :

(-2.82 -0.15 -8.05 -9.91 8.85)

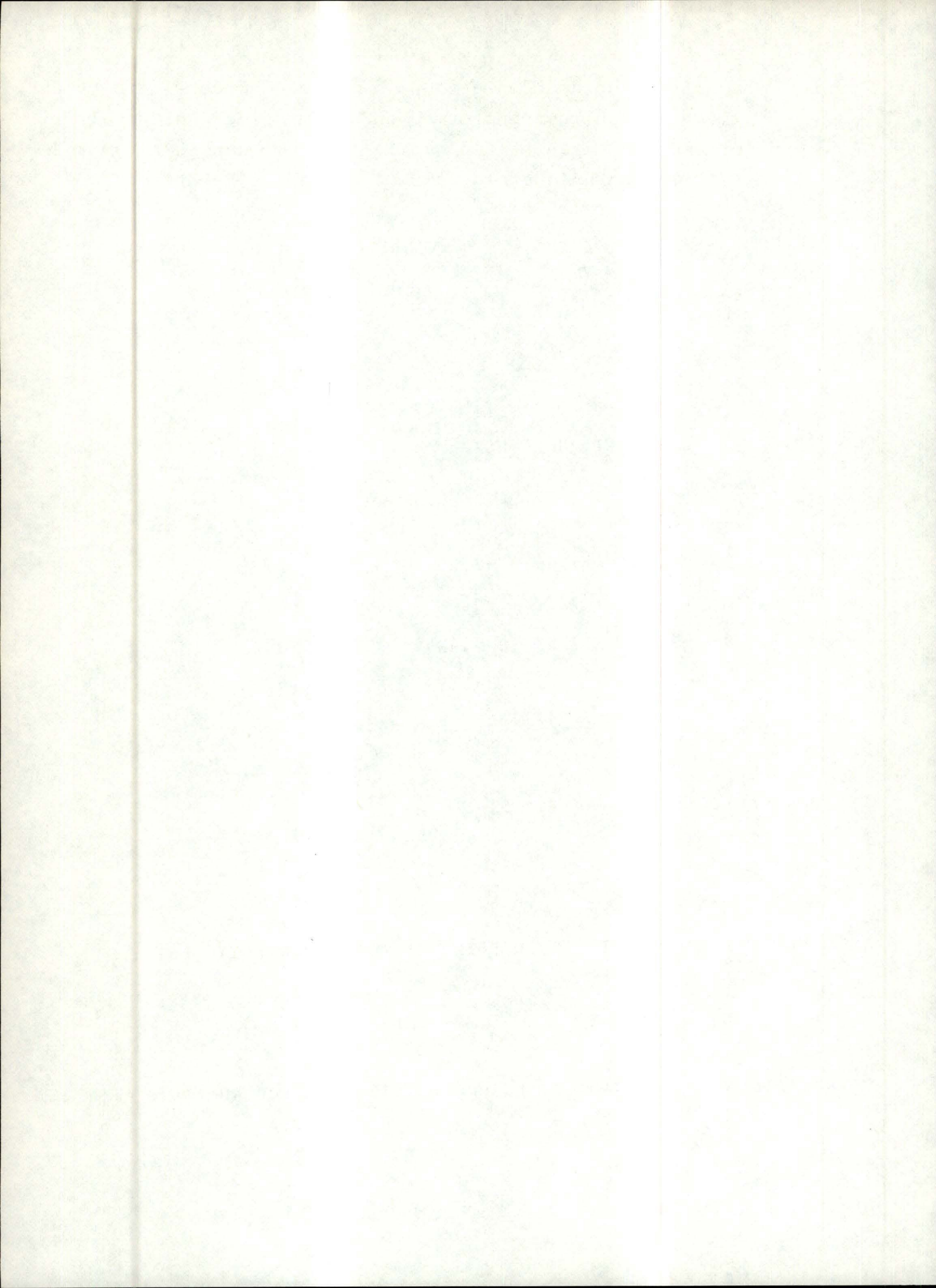
LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(- - - - +)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERO D EST :

< 4 >

L'INDICE 4 S'ETANT AJOUTE



LE TABLEAU DE L'ITERATION 1 EST :

	Z01	Z02	Z03	W04	Z05	Q
W01	35.98	-7.91	6.72	-0.26	6.73	-5.35
W02	-4.74	19.18	0.94	-0.02	9.34	-0.32
W03	-5.76	-2.42	19.46	0.23	-2.95	-5.81
Z04	-0.25	0.19	-0.14	0.04	-0.19	0.35
W05	-8.19	-5.42	-7.26	-0.12	35.32	7.70

LE POINT DE COMPLEMENTARITE EST :

(-5.35 -0.32 -5.81 0.35 7.70)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(- - - + +)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERO 1 EST :

< 4 3 >

L'INDICE 3 S'ETANT AJOUTE

LE TABLEAU DE L'ITERATION 2 EST :

	Z01	Z02	W03	W04	Z05	Q
W01	37.97	-7.08	0.35	-0.33	7.75	-3.34
W02	-4.46	19.30	0.05	-0.03	9.49	-0.04
Z03	0.30	0.12	0.05	-0.01	0.15	0.30
Z04	-0.29	0.18	-0.01	0.04	-0.22	0.31
W05	-10.34	-6.32	-0.37	-0.03	34.22	5.54

LE POINT DE COMPLEMENTARITE EST :

(-3.34 -0.04 0.30 0.31 5.54)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(- - + + +)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERO 2 EST :

< 4 3 1 >

L'INDICE 1 S'ETANT AJOUTE

LE TABLEAU DE L'ITERATION 3 EST :

	W01	Z02	W03	W04	Z05	Q
Z01	0.03	0.19	-0.01	0.01	-0.20	0.09
W02	-0.12	18.47	0.09	-0.07	10.40	-0.43
Z03	0.01	0.18	0.05	-0.01	0.09	0.32
Z04	-0.01	0.12	-0.00	0.03	-0.16	0.28
W05	-0.27	-8.25	-0.28	-0.12	36.33	4.63

LE POINT DE COMPLEMENTARITE EST :

(0.09 -0.43 0.32 0.28 4.63)

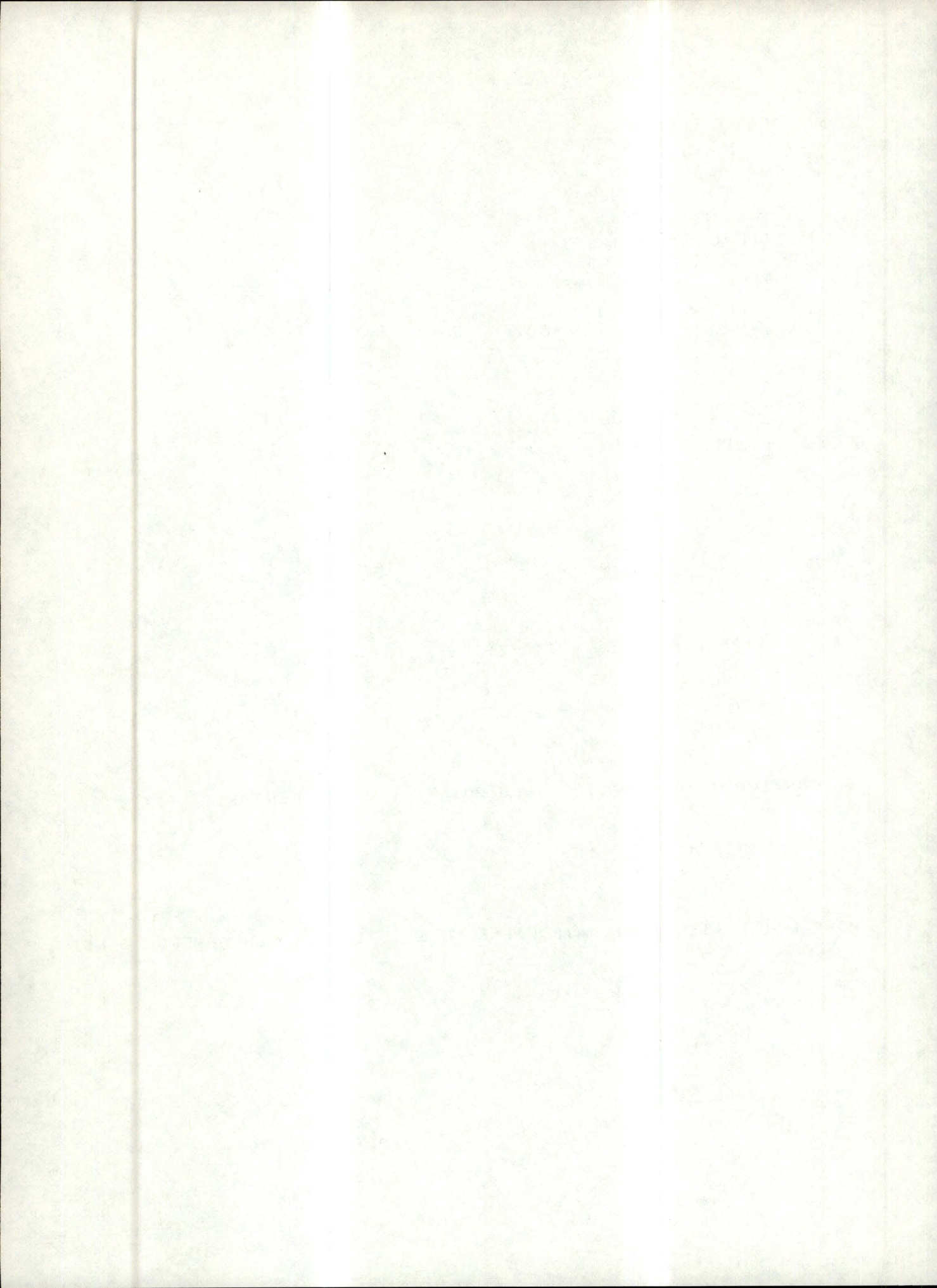
LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(+ - + + +)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERO 3 EST :

< 4 3 1 2 >

L'INDICE 2 S'ETANT AJOUTE



LE TABLEAU DE L'ITERATION 4 EST :

	W01	W02	W03	W04	Z05	G
Z01	0.03	0.01	-0.01	0.01	-0.31	0.09
Z02	0.01	0.05	-0.00	0.00	-0.56	0.02
Z03	0.01	0.01	0.05	-0.01	-0.01	0.33
Z04	-0.01	0.01	-0.01	0.03	-0.22	0.28
W05	-0.32	-0.45	-0.24	-0.15	40.98	4.43

LE POINT DE COMPLEMENTARITE EST :

(0.09 0.02 0.33 0.28 4.43)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(+ + + + +)

LE PT. DE COMPL., SOLUTION DU PROBLEME EST :

(0.09 0.02 0.33 0.28 4.43)

B Algorithme de Murty

La description et l'organigramme de cet algorithme sont tout-à-fait analogues à ceux de Bard. Le seul changement se situe au niveau de la recherche d'une composante négative du point complémentaire.

Pour l'algorithme de Bard, nous recherchions la composante la plus négative du vecteur. Dans le cas de l'algorithme de Murty, nous rechercherons la première composante négative du point complémentaire. Nous remplacerons donc, dans l'organigramme, l'assertion :

$$\text{" } \min \{ \lambda_j^k \}_{j \in N} = \lambda_i^k \text{" par " } \min \{ j \mid \lambda_j^k < 0 \} = \lambda_i^k$$

L'algorithme de Murty est donc, lui aussi, un algorithme direct de pivot unique.

Exemple numérique : Dans cet exemple, nous poserons $y = f(x)$

$$\text{avec } f: \mathbb{R}^2 \longrightarrow \mathbb{R}^2$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \longmapsto \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} +3 + 3x_1 + 2x_2 \\ -2 + 3x_1 + x_2 \end{pmatrix}$$

$$1) I_0 = \emptyset$$

Dès lors, le système à résoudre est :

$$\begin{cases} x_1 = 0 \\ x_2 = 0 \\ y_1 = 3 + 3x_1 + 2x_2 \\ y_2 = -2 + 3x_1 + x_2 \end{cases}$$

La solution immédiate est $\lambda_0 = (3, -2)$

L'ensemble d'indexation suivant est donc : $I_\lambda = \emptyset \cup \{2\} = \{2\}$.

$$2) \mathcal{I}_\lambda = \{2\}$$

Dès lors, le système à résoudre est :

$$\begin{cases} \alpha_1 = 0 \\ \alpha_2 = 0 \\ y_1 = 3 + 3\alpha_1 + 2\alpha_2 \\ y_2 = -2 + 3\alpha_1 + \alpha_2 \end{cases}$$

Ce qui conduit à un échange de variables x_2 et y_2 et donc à un pivotage sur l'élément 1.

	1	x_1	x_2
y_1	3	3	2
y_2	-2	3	1
	1	x_1	y_2
y_1	7	-3	2
x_2	2	-2	1

La solution est : $\lambda_1 = (7, 2)$

Elle est positive; donc la solution du problème est :

$$(\alpha_1, \alpha_2, y_1, y_2) = (0, 2, 7, 0).$$

Exemple programmé de dimension 4 dans le cas linéaire :

Il convient de faire les mêmes remarques que pour l'exemple programmé de l'algorithme de Bard.

METHODE DE MURTY

LE TABLEAU DE L'ITERATION 0 EST :

	Z01	Z02	Z03	Z04	Q
W01	19.93	-7.27	2.29	-0.82	-5.59
W02	-6.14	19.79	-3.64	-3.54	-8.51
W03	0.80	1.37	9.33	-6.12	-6.09
W04	-1.41	6.29	-9.53	23.46	3.08

LE POINT DE COMPLEMENTARITE EST :

(-5.59 -8.51 -6.09 3.08)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(- - - +)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERIC 0 EST :

< 1 >

L'INDICE 1 S'ETANT AJOUTE

LE TABLEAU DE L'ITERATION 1 EST :

	W01	Z02	Z03	Z04	Q
Z01	0.05	0.36	-0.12	0.04	0.28
W02	-0.31	17.55	-2.93	-3.79	-10.23
W03	0.04	1.66	9.24	-6.09	-5.86
W04	-0.07	5.73	-9.36	23.40	2.68

LE POINT DE COMPLEMENTARITE EST :

(0.28 -10.23 -5.86 2.68)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(+ - - +)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERO 1 EST :

< 1 2 >

L'INDICE 2 S'ETANT AJOUTE

LE TABLEAU DE L'ITERATION 2 EST :

	W01	W02	Z03	Z04	Q
Z01	0.06	0.02	-0.05	0.12	0.49
Z02	0.02	0.06	0.17	0.22	0.58
W03	0.07	0.09	9.52	-5.73	-4.89
W04	0.03	0.33	-8.40	24.65	6.05

LE POINT DE COMPLEMENTARITE EST :

(0.49 0.58 -4.89 6.05)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(+ + - +)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERO 2 EST :

< 1 2 3 >

L'INDICE 3 S'ETANT AJOUTE

LE TABLEAU DE L'ITERATION 3 EST :

	W01	W02	W03	Z04	Q
Z01	0.06	0.02	-0.01	0.09	0.47
Z02	0.02	0.06	0.02	0.32	0.67
Z03	-0.01	-0.01	0.11	0.66	0.51
W04	0.09	0.41	-0.88	19.59	1.73

LE POINT DE COMPLEMENTARITE EST :

(0.47 0.67 0.51 1.73)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(+ + + +)

LE PT. DE COMPL., SOLUTION DU PROBLEME EST :

(0.47 0.67 0.51 1.73)

3

ALGORITHME DE M. KOSTREVAA Préliminaire portant sur la théorie des groupes

Soient $A = \{ 1, 2, 3, \dots, 2^n \}$,

$T = \{ \Phi \mid \Phi \}$ est une permutation de A sur A .

Nous savons que (T, \circ) forme un groupe : le groupe symétrique d'ordre 2^n .

Définitions : 1) Si $j \in A$ et $\Phi \in T$,
alors, l'"orbite de nombre j pour Φ " est définie par les nombres : $j, \Phi(j), \Phi^2(j), \dots, \Phi^{\ell-1}(j)$
où ℓ est le plus petit entier positif tel que $\Phi^\ell(j) = j$

2) Le nombre ℓ est appelé "longueur de l'orbite".

Lemme 3.1. : Les nombres définissant une orbite sont tous différents.

Démonstration : Soient $A = \{ 1, 2, \dots, 2^m \}$
 $\Phi \in T$
 $\alpha, \Phi(\alpha), \Phi^2(\alpha), \dots, \Phi^{\ell-1}(\alpha)$ l'orbite de α

La démonstration se fait par l'absurde.

Supposons que, dans l'orbite L , i soit le plus petit indice tel qu'il existe $j < i$ avec $\Phi^i(\alpha) = \Phi^j(\alpha)$
(autrement dit, l'élément $\Phi^j(\alpha)$ se trouve répété en $\Phi^i(\alpha)$).

Comme (T, \circ) est un groupe, nous avons :

$$\Phi^{-1} \Phi^j(\alpha) = \Phi^{-1} \Phi^i(\alpha)$$

$$\Phi^{j-1}(\alpha) = \Phi^{i-1}(\alpha)$$

Ceci est contradictoire car i est le premier indice pour lequel

$$\Phi^i(x) = \Phi^i(x)$$

Donc, l'orbite L est telle que $\Phi^l(x) = x$.

Remarque : Ce lemme est important car il est fondamental dans la preuve de la convergence de l'algorithme de M. Kostreva.

B Les données de l'algorithme et leur forme

Afin de pouvoir débiter la résolution d'un cpf par la méthode de Kostreva, nous devons nous donner au préalable

- des bijections

$$\begin{aligned} \epsilon: A = \{1, 2, \dots, 2^m\} &\longrightarrow \mathcal{P}(N) \text{ tel que } \epsilon(1) = \emptyset \\ k &\longmapsto \epsilon(k) \text{ noté } I_k \\ &(\emptyset = \text{ensemble vide}) \end{aligned}$$

$$\begin{aligned} \nu: A &\longrightarrow \oplus \\ k &\longmapsto \nu(k) \text{ noté } \oplus_k \end{aligned}$$

- une permutation Φ des nombres de $\{1, 2, \dots, 2^n\}$

$$\Phi: A \longrightarrow A$$

Ce que nous résumons facilement par le tableau suivant :

<u>Numéros (de 1 à 2^m)</u>	<u>Ensemble d'indexation</u>	<u>Orthant</u>
1	$t(1) = I_1 = \phi$	$r(1) = \textcircled{+}_1 = \mathbb{R}_+^m$
2	.	.
3	.	.
.	.	.
.	.	.
.	.	.
k	$t(k) = I_k$	$r(k) = \textcircled{+}_k$
.	.	.
.	.	.
2^m	$t(2^m) = I_{2^m}$	$r(2^m) = \textcircled{+}_{2^m}$

Le choix des ensembles d'indexation et des orthants est quelconque, sous réserve que l'ensemble d'indexation " ϕ " et l'orthant " \mathbb{R}_+^m " correspondent au même numéro α .

Cet algorithme consiste, comme pour les algorithmes précédents, à résoudre un système à partir d'un certain ensemble d'indexation. Chaque itération de l'algorithme effectue une itération (qui est la composée de bijections) sur l'ensemble A .

C Description et organigramme

Hypothèse : Soit f une fonction dont la matrice jacobienne est bornée positivement (c'est-à-dire $f \in \text{PBJM}$).

Description :

Pas 1 : Initialement : choix des données :

- les bijections

- la permutation

$$l = 1$$

Pas 2 : On effectue la permutation en calculant

$$k = \phi(l)$$

aller au pas 3.

Pas 3 : A partir du tableau des données, on détermine l'ensemble d'indexation $I_k = t(k)$ (trivial).
Aller au pas 4.

Pas 4 : On résoud le système :

$$\left\{ \begin{array}{ll} y = f(x) & \\ x_i^k = 0 & \text{si } i \notin I_k \\ f_i(x^k) = 0 & \text{si } i \in I_k \end{array} \right.$$

Ce qui nous permettra, au pas 5, de déterminer l'unique point complémentaire $\lambda^{(k)}$ (unicité résulte de l'hypothèse PBJM).

Pas 5 : On détermine le point complémentaire :

$$\lambda_i^{(k)} = x_i^k + f_i(x^k)$$

Aller au pas 6.

Pas 6 : A partir du tableau des données, on détermine l'orthant Θ_l correspondant au point complémentaire $p^{(k)}$ (Θ_l est l'orthant déterminé à l'aide de la configuration de signes du point complémentaire).

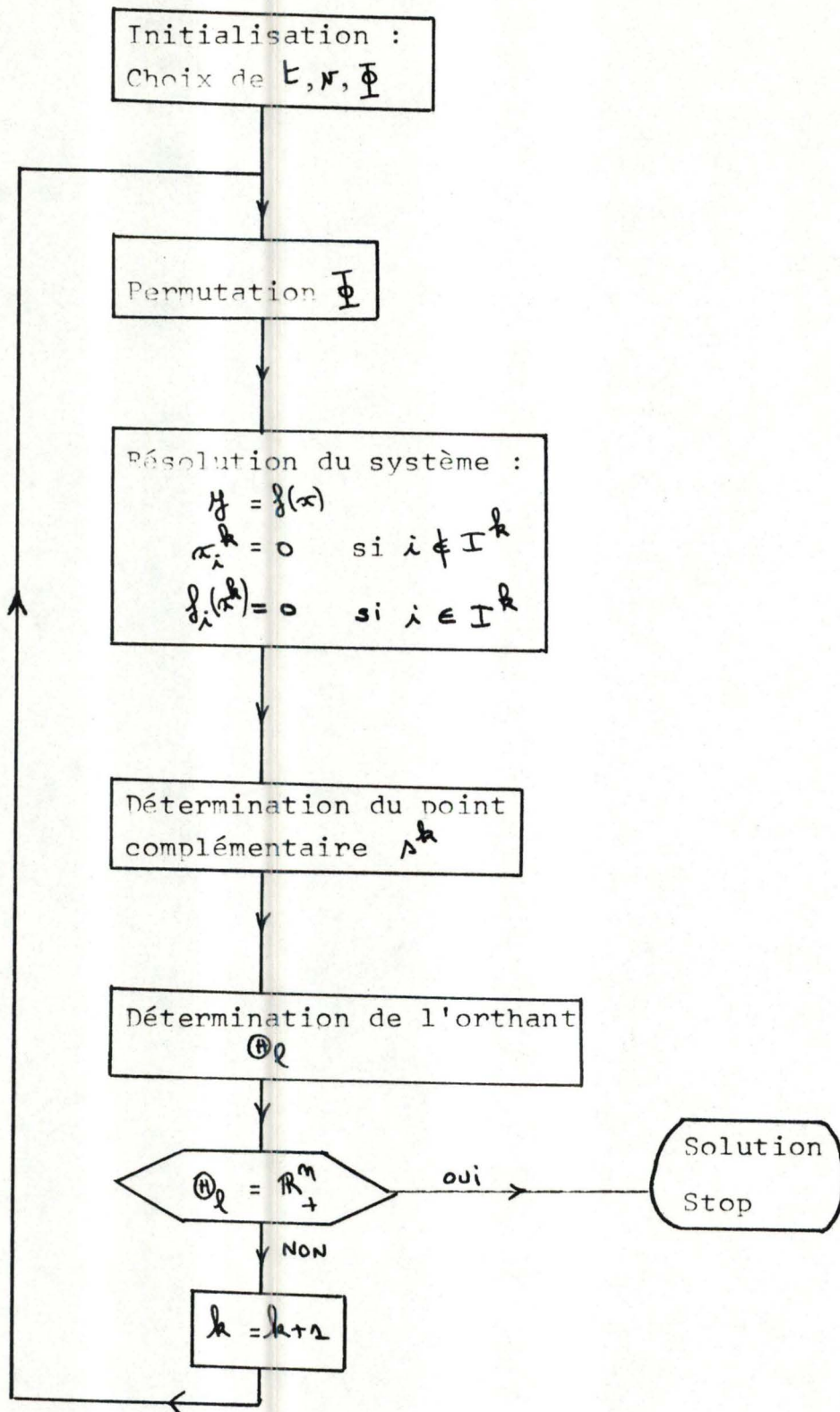
$$\Theta_l = (g \circ S)(l).$$

Aller au pas 7.

Pas 7 : Si $\Theta_l = \mathbb{R}_+^m$, c'est-à-dire si $l = 1$, $p^{(k)}$ est la solution du problème; on s'arrête.
Si non, aller au pas 8.

Pas 8 : On vient d'accomplir une itération complète; on passe à l'itération suivante : $k = k + 1$
Aller au pas 2.

Organigramme



D Convergence de l'algorithmeThéorème 3.1.

Soit une fonction PBJM sur \mathbb{R}^n .

Supposons que le problème cpf considéré est non-dégénéré.

Alors, pour toutes bijections Φ, τ, κ telles que l'ensemble ϕ et l'orthant \mathbb{R}_+^n , correspondent au même numéro α , l'algorithme décrit ci-dessus converge.

Démonstration : La démonstration de ce théorème se décompose, pour arriver à son terme, en deux parties.

La première démontrera qu'une itération complète de l'algorithme effectue une permutation sur les nombres de 1 à 2^n .

La seconde montrera que l'indice 1 est le premier indice qui se répétera.

Ces deux parties se complètent pour montrer que σ fournit un algorithme direct convergent.

Première partie : Une itération complète de l'algorithme effectue une permutation sur les numéros.

Démonstration :

Soient Φ, S, g, ψ, τ des fonctions telles que :

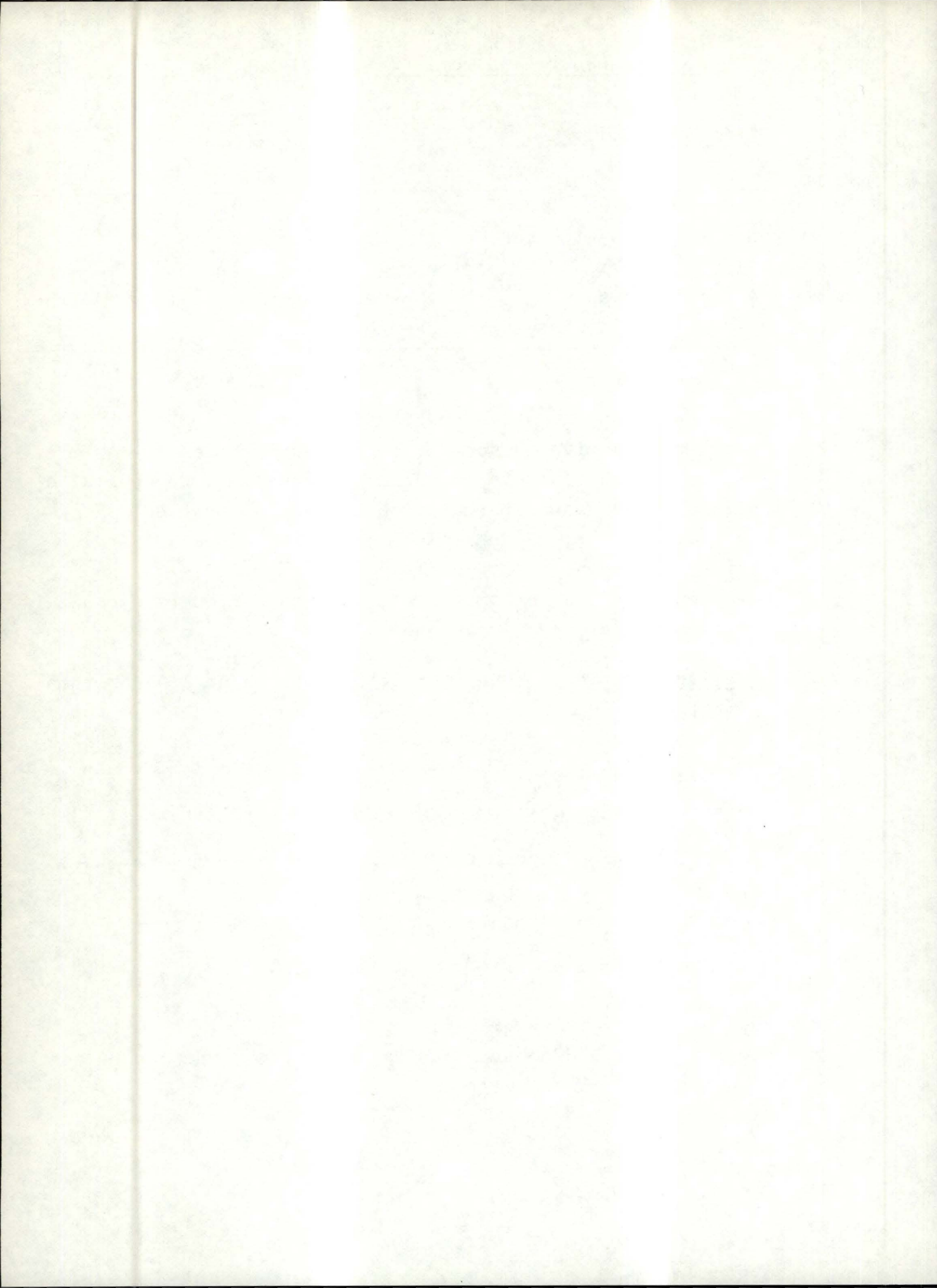
$$A \xrightarrow{\Phi} A \xrightarrow{\tau} I \xrightarrow{S} X \xrightarrow{g} \mathbb{H} \xrightarrow{\kappa^{-1}} A$$

Nous devons donc démontrer que la fonction

$$\sigma = \phi \circ \tau \circ S \circ g \circ \kappa^{-1} : A \longrightarrow A$$

est une application bijective.

Ceci revient à démontrer que Φ, τ, S, g, ψ sont toutes des applications bijectives.



- 1 - La fonction \mathbb{I} est trivialement une application bijective, car, par hypothèse, elle est une permutation.
- 2 - La fonction \mathbb{E} est trivialement une application bijective, car, dans le tableau initial des données, on associe à chaque indice k un, et un seul, ensemble d'indexation.
- 3 - Nous avons montré au chapitre 2 que, par la propriété de "non dégénérescence", la fonction S est aussi une application bijective.
- 4 - Egalement dans le chapitre 2, nous avons montré que par le principe du "trou du pigeon" la fonction g est une application bijective.
- 5 - La fonction \mathcal{N}^{-1} est trivialement une application bijective, car \mathcal{N} l'était.

Ainsi donc, la composée de ces différentes applications bijectives est une application bijective. Donc, la fonction $\sigma : A \rightarrow A$ est une permutation.

Deuxième partie : Le nombre 1 est le premier nombre de l'orbite qui, par la permutation σ , se répète.

Démonstration :

Par la première partie, l'application σ est une permutation. Par le lemme précédent, tous les nombres définissant l'orbite sont différents.

Ainsi donc, il faudra effectuer au plus 2^n permutations pour que l'indice 1 se répète le premier.

C'est pourquoi, l'algorithme de M. Kostreva se termine en un nombre fini de pas.

Remarque :

Si nous reprenons l'organigramme d'un algorithme direct, nous constatons que cet algorithme est un algorithme direct de type "pivot-bloc".

E EXEMPLES NUMERIQUES

Dans ces exemples, nous posons :

1°) Dans \mathbb{R}^2

$$f: \mathbb{R}^2 \rightarrow \mathbb{R}^2: \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \longrightarrow \begin{pmatrix} 2x_1 + x_2 + 1 \\ -x_1 + x_2 - 1 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

$$A = \{1, 2, 3, 4\}$$

$$\Phi: A \rightarrow A: i \longrightarrow \begin{cases} i+1 & \forall i < 4 \\ 1 & \text{sinon} \end{cases}$$

$$A \xrightarrow{\tau} I \xrightarrow{\Phi} \oplus$$

1	∅	(+ +)
2	{1}	(+ -)
3	{2}	(- +)
4	{1, 2}	(- -)

1ère itération

	1	x_1	x_2
v_1	1	2	1
v_2	-1	-1	①

$$I^0 = \emptyset$$

$$s = (1, -1)$$

$$\oplus = (+ -)$$

2ème itération

	1	x_1	v_2
v_1	3	3	1
x_2	1	1	1

$$I^1 = \{2\}$$

$$s = (3, 1)$$

$$\oplus = (+ +) \Rightarrow \text{terminé}$$

2°) Dans \mathbb{R}^3

$$f : \mathbb{R}^3 \rightarrow \mathbb{R}^3 : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \rightarrow \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} x_1 - x_2 + x_3 + 1 \\ x_1 + x_2 - x_3 - 1 \\ -x_1 + x_2 + 2x_3 + 1 \end{pmatrix}$$

$$A = \{1, 2, \dots, 8\}$$

A	\xrightarrow{E}	A	$\xrightarrow{g \circ S}$	\oplus
1		\emptyset		(+ + +)
2		{1}		(+ + -)
3		{2}		(+ - +)
4		{3}		(- + +)
5		{1, 2}		(- + -)
6		{1, 3}		(- - +)
7		{2, 3}		(+ - -)
8		{1, 2, 3}		(- - -)

1ère itération

	1	x_1	x_2	x_3	
v_1	1	1	-1	1	$I^0 = \emptyset$
v_2	-1	1	1	-1	$S^0 = (1 \ -1 \ 1)$
v_3	1	-1	1	②	$\oplus = (+ \ - \ +)$

2ème itération

	1	x_1	x_2	y_3	
v_1	1/2	③/2	-3/2	1/2	$I^1 = \{3\}$
v_2	-1/2	1/2	3/2	-1/2	$S^1 = (1/2 \ -1/2 \ -1/2)$
x_3	-1/2	1/2	-1/2	1/2	$\oplus = (+ \ - \ -)$

3ème itération

	1	v_1	x_2	v_3
x_1	$-1/3$	1	1	$-1/3$
v_2	$-2/3$	$1/3$	①	$-1/3$
x_3	$-2/3$	$1/3$	-1	$2/3$

$$I^2 = \{1, 2, 3\}$$

$$s^2 = (1/3 \ 1 \ 1)$$

$$\oplus = (+ \ + \ +)$$

→ terminé

	1	v_1	v_2	v_3
x_1	$1/3$	$2/3$	1	0
x_2	$2/3$	$-1/3$	1	$1/3$
x_3	$-1/3$	$2/3$	-1	① $1/3$

	1	v_1	v_2	x_3
x_1	$1/3$	$2/3$	1	0
x_2	1	-1	2	1
v_3	1	-2	3	1

F Exemple montrant la nécessité de l'hypothèse : "non dégénérescence"

Dans cet exemple, nous posons :

$$f : \mathbb{R}^3 \rightarrow \mathbb{R}^3 : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \rightarrow \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} x_1 - x_2 + x_3 + 1 \\ x_1 + x_2 - x_3 - 1 \\ -x_1 + x_2 + x_3 + 1 \end{pmatrix}$$

$$A = \{1, 2, \dots, 8\}$$

$$\Phi : J \rightarrow J : i \rightarrow \begin{cases} i+1 & \text{si } i < 8 \\ 1 & \text{sinon} \end{cases}$$

L'application $g_{\circ S}$ est la même que pour l'exemple précédent

1ère itération

	1	x_1	x_2	x_3
v_1	1	1	-1	1
v_2	-1	1	1	-1
v_3	1	-1	1	1

$$I^0 = \emptyset$$

$$s^0 = (1 \ -1 \ 1)$$

$$\Theta^0 = (+ \ - \ +)$$

2ème itération

	1	x_1	v_2	x_3
v_1	0	2	-2	1
x_2	0	0	2	-1
v_3	-1	1	-1	1

$$I^1 = \{2\}$$

$$s^1 = (0 \ 0 \ -1)$$

$$\Theta^1 = (+ \ + \ -)$$

3ème itération

	1	x_1	v_2	y_3
v_1	0	2	-1	0
x_2	0	0	1/2	1/2
x_3	-1	1	-1/2	1/2

$$I^2 = \{2\} \Rightarrow \text{on cycle !}$$

G Remarques

1) Cet algorithme peut être facilement modifié pour résoudre un GCP pour $K = \oplus$, un orthant quelconque de \mathbb{R}^m . Il suffit donc pour le résoudre que l'ensemble d'indexation Φ et l'orthant \oplus soient associés au même numéro (1 par exemple).

2) Si on prend comme permutation, l'application identité, l'ensemble d'indexation $I^{(k+1)}$ est déterminé très facilement.

$$I^{(k+1)} = \{i \mid \Delta_i^{(k)} < 0\}$$

3) Si la fonction de départ est du type :

$$f(x) = (f_1(x_1), \dots, f_m(x_m))^T$$

alors l'algorithme converge en une seule étape et la configuration de signe de $f(0)$ s'identifie automatiquement à l'ensemble d'indexation optimum I_0^* .

Exemple :

$$f : \mathbb{R}^3 \rightarrow \mathbb{R}^3 : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \rightarrow \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} -2+x_1 \\ -3+x_2 \\ 1+2x_3 \end{pmatrix}$$

$$I = \mathbb{1}_{\mathbb{R}^3}$$

$$I_0 = \emptyset \quad s^0 = (-2 \ 3 \ 1)$$

$$\ominus^0 = (- \ - \ +) \Rightarrow I_1 = \{1, 2\}$$

$$I_1 = \{1, 2\} \Rightarrow \begin{cases} x_1 = 2 + y_1 \\ x_2 = 3 + y_2 \\ x_3 = 1 + 2y_3 \end{cases}$$

$$\Rightarrow s^* = (2 \ 3 \ 1)$$

$$\oplus^* = (+ \ + \ +) \Rightarrow \text{terminé}$$

H Exemple programmé

Il convient de faire les mêmes remarques que pour l'exemple programmé des algorithmes de Bard et de Murty.

METHODE DE KOSTREVA

0	< >	(+ + + +)
1	< 4 >	(+ + + -)
2	< 3 >	(+ + - +)
3	< 3 4 >	(+ + - -)
4	< 2 >	(+ - + +)
5	< 2 4 >	(+ - + -)
6	< 2 3 >	(+ - - +)
7	< 2 3 4 >	(+ - - -)
8	< 1 >	(- + + +)
9	< 1 4 >	(- + + -)
10	< 1 3 >	(- + - +)
11	< 1 3 4 >	(- + - -)
12	< 1 2 >	(- - + +)
13	< 1 2 4 >	(- - + -)
14	< 1 2 3 >	(- - - +)
15	< 1 2 3 4 >	(- - - -)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERO 0 EST :

< >

LE (LES) TABLEAU(X) DE L'ITERATION 0 EST(SONT) :

	Z01	Z02	Z03	Z04	Q
W01	19.93	-7.27	2.29	-0.82	-5.59
W02	-6.14	19.79	-3.64	-3.54	-8.51
W03	0.80	1.37	9.33	-6.12	-6.09
W04	-1.41	6.29	-9.55	23.46	3.08

LE POINT DE COMPLEMENTARITE EST :

(-5.59 -8.51 -6.09 3.08)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(- - - +)

POUR L'ITERATION SUIVANTE, L'ENSEMBLE D'INDEXATION NUMERO 0 EST :

< 1 2 3 >

	W01	Z02	Z03	Z04	Q
Z01	0.05	0.36	-0.12	0.04	0.28
W02	-0.31	17.55	-2.93	-3.79	-10.23
W03	0.04	1.66	9.24	-6.09	-5.86
W04	-0.07	5.78	-9.36	23.40	2.68

	W01	W02	Z03	Z04	Q
Z01	0.06	0.02	-0.05	0.12	0.49
Z02	0.02	0.06	0.17	0.22	0.58
W03	0.07	0.09	9.52	-5.73	-4.89
W04	0.03	0.33	-8.40	24.65	6.05

	W01	W02	W03	Z04	Q
Z01	0.06	0.02	-0.01	0.09	0.47
Z02	0.02	0.06	0.02	0.32	0.67
Z03	-0.01	-0.01	0.11	0.60	0.51
W04	0.09	0.41	-0.88	19.59	1.73

LE POINT DE COMPLEMENTARITE EST :

(0.47 0.67 0.51 1.73)

LA CONFIGURATION DE SIGNES DU POINT DE COMPLEMENTARITE EST :

(+ + + +)

LE PT. DE COMPL., SOLUTION DU PROBLEME EST :

(0.47 0.67 0.51 1.73)

C H A P I T R E I V

- - - - -

COMPARAISONS SUCCINTES DES TROIS ALGORITHMES PRESENTES

=====

INTRODUCTION

=====

Dans ce mémoire, nous avons rencontré deux types d'algorithmes directs de type pivot unique illustrés par les schémas de Bard et de Murty et les algorithmes directs de types pivot bloc illustrés par l'algorithme de Kostreva. Ces deux catégories d'algorithmes se différencient l'une de l'autre par le choix de l'ensemble d'indexation. Celui-ci est déterminé, pour Bard et Murty, à partir d'un test de minimisation sur les composantes du point complémentaire. Dans le cas de l'algorithme de Kostreva, le choix de l'ensemble d'indexation est basé sur les données d'une permutation, dans l'ensemble $A = \{1, 2, \dots, 2^n\}$ | n est la dimension du problème | et de bijections

$$\tau : A \xrightarrow{+} \mathcal{P}(N) \quad \text{et} \quad \nu : A \xrightarrow{+} \oplus.$$

Ces différences fondamentales suggèrent plusieurs questions :

- 1° Lequel des algorithmes de Bard et Murty est-il le plus efficace du point de vue du temps d'exécution ?
- 2° Comment comparer l'algorithme de Kostreva par rapport aux deux autres ?
- 3° Le choix de permutations différentes dans les données de l'algorithme de Kostreva, entraîne-t-il des variations de la rapidité de convergence ?

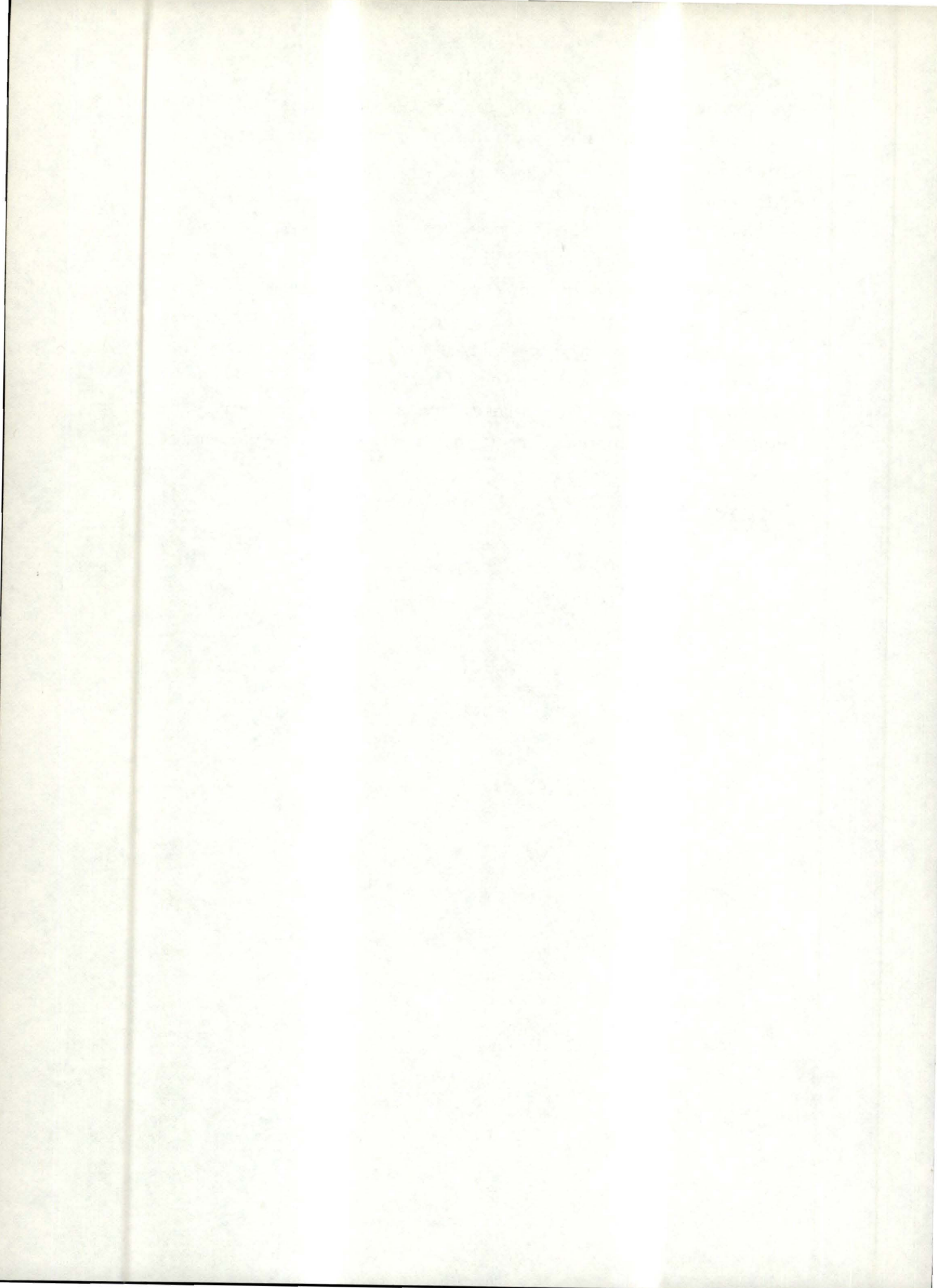
Afin d'analyser ces différentes questions, ces trois algorithmes ont été programmés en fortran IV pour l'ordinateur Siemens 4004 et ont été testés, chacun, sur une même série de problèmes linéaires de complémentarité dont les données

ont été générées aléatoirement.

Nous nous intéresserons spécialement, dans chacun des cas, à la comparaison du nombre de pivotages car cette opération est identique dans les trois algorithmes et a été programmée de manière identique.

Dans notre cas, établir une comparaison des nombres d'itérations ne présente pas d'intérêt particulier car nous ne pouvons comparer que des opérations de même type et le concept d'itération est fort différent pour les algorithmes de pivot unique (un pivotage par itération) et pour ceux de pivot bloc (plusieurs pivotages par itération).

Le temps d'exécution étant pratiquement proportionnel au nombre de pivotages, nous nous limiterons à comparer les nombres de ceux-ci.



1 PROGRAMMATION
 =====

A Algorithme de Bard

PROGRAM BARD

TABLE DES VARIABLES

AUX : MATRICE NXN DES DONNEES
 Q : VECTEUR POINT COMPLEMENTAIRE
 INDEX : ENSEMBLE D'INDEXATION
 KMIN : INDICE DE LA PREMIERE COMPOSANTE NEGATIVE DU POINT
 COMPLEMENTAIRE
 M : DIMENSION DE LA MATRICE DES DONNEES
 N1 : DIMENSION TOTALE DES DONNEES
 R : INDICE DE LA LIGNE DU PIVOT
 T : INDICE DE LA COLONNE DU PIVOT
 NUMIT : COMPTEUR DU NOMBRE D'ITERATIONS
 SIGN : VECTEUR LOGIQUE DE LA CONFIGURATION DE SIGNES DU
 POINT COMPLEMENTAIRE
 CAR : VECTEUR LOGIQUE COMPOSE DES DEUX VARIABLES + ET -
 TOUTES LES AUTRES VARIABLES SONT DES VARIABLES INTERMEDIAIRES

IMPLICIT INTEGER(Q-Z)
 IMPLICIT REAL *8(A-B,P)
 LOGICAL *1 SIGN(10), CAR(2)
 INTEGER AZ
 INTEGER *2 BZ(2), EZ
 INTEGER FMT1 *2 (14), FMT2 *2 (13), FMT4 *2 (13)
 DIMENSION Z(11), W(10), INDEX(10)
 COMMON AUX(10,11), N, N1
 EQUIVALENCE (AZ, BZ(1))
 DATA QAUX/' Q'/
 DATA FMT1/'(1H+,10X,3X,00(F6.2,3X),1H))'/
 DATA FMT2/'(1H+,10X,3X,00(A1,3X),1H))'/
 DATA FMT4 /'(1H+,10X,3X,00(I2,3X),1H>)/
 DATA E7/' Z'/
 READ 1000,N
 40 1000 FORMAT(I2)

INITIALISATION

NUMIT=J
 MPI=1
 MI=0
 N1=N+1
 NAUX=N/10

```

51      NA=N-10*NAUX
52      CALL GENER
53
54
55 C          CONSTRUCTION DU TABLEAU DES DONNEES
56
57
58      READ 1002,(Z(I),I=1,N)
59      READ 1002,(W(I),I=1,N)
60 1002      FORMAT(10A4)
61      READ 1003,(CAR(I),I=1,2)
62 1003      FORMAT(2A1)
63          Z(N1)=GAUX
64          PRINT 2004
65 2004      FORMAT(1H1,////,12X,'METHODE DE MURTY')
66          PRINT 2002,NUMIT
67 2002      FORMAT(////////,1X,'LE TABLEAU DE L'ITERATION',13,' EST :')
68          PRINT 2003,(Z(I),I=1,N1)
69 2003      FORMAT(///,5X,11(A6,2X))
70          DO 3 K=1,N
71 3          PRINT 2005,W(K),(AUX(K,I),I=1,N1)
72 2005      FORMAT(1H),44,11(1X,F6.2,1X))
73
74
75 C          PREMIERE ETAPE DE L'ALGORITHME :
76 C          ON RECHERCHE - LE POINT COMPLEMENTAIRE
77 C                      - SA CONFIGURATION DE SIGNES
78 C          ON VERIFIE SA POSITIVITE
79 C          ON DEDUIT L'ENSEMBLE D'INDEXATION QUI NOUS PERMET
80 C          DE TROUVER L'ELEMENT PIVOT
81
82
83          FMT1(7)=FMT1(7)+256*NAUX+NA
84          PRINT 2000
85 2000      FORMAT(///,1X,'LE POINT DE COMPLEMENTARITE EST :',///,10X,'(')
86          WRITE(6,FMT1)(AUX(I,N1),I=1,N)
87          CALL SIGNE(SIGN,CAR)
88          FMT2(7)=FMT2(7)+256*NAUX+NA
89          PRINT 2001
90 2001      FORMAT(///,1X,'LA CONFIGURATION DE SIGNES DU POINT '
91 C'DE COMPLEMENTARITE EST :',///,10X,'(')
92          WRITE(6,FMT2)(SIGN(I),I=1,N)
93          CALL POSIT(NAUX,NA)
94          CALL AMIN(KMIN)
95          INDEX(1)=KMIN
96          R=KMIN
97          T=KMIN
98          FMT4(7)=FMT4(7) + 1
99          PRINT 2006,NUMIT
100 2006      FORMAT(///,1X,'POUR L'ITERATION SUIVANTE,L'ENSEMBLE '

```

```

101      C, ' D'INDEXATION NUMERO ',I2,' EST :', ///,10X,'<' )
102      WRITE(6,FMT4) KMIN
103      PRINT 2008,KMIN
104 2008  FORMAT(//,1X,' L'INDICE',I2,' S'ETANT AJOUTE')
105 2009  FORMAT(//,1X,' L'INDICE',I2,' S'ETANT RETIRE')
106
107
108 C          IE ME ETAPE DE L'ALGORITHME
109 C          ON EFFECTUE UN PIVOTAGE
110 C          ON RECHERCHE - LE POINT COMPLEMENTAIRE
111 C          - SA CONFIGURATION DE SIGNES
112 C          ON VERIFIE SA POSITIVITE
113 C          ON DEDUIT L'ENSEMBLE D'INDEXATION SUIVANT
114
115
116 100      CALL PIVOT(R,T,NUMIT,Z,W)
117      PRINT 2000
118      WRITE(6,FMT1)(AUX(I,N1),I=1,N)
119      CALL SIGNE(SIGN,CAR)
120      PRINT 2001
121      WRITE(6,FMT2)(SIGN(I),I=1,N)
122      CALL POSIT(NAUX,NA)
123      CALL AMIN(KMIN)
124      AZ=W(KMIN)
125      IF(BZ(1).EQ.EZ) GOTO 8
126      MPI=MPI+1
127      MP=MPI-MI
128      INDEX(MP)=KMIN
129      PRINT 2006,NUMIT
130      NAUX=MP/10
131      MA=MP-10*NAUX
132      FMT4(7)=15*16 + 15*16*#3
133      FMT4(7)=FMT4(7)+256*NAUX+MA
134      WRITE(6,FMT4)(INDEX(K),K=1,MP)
135      PRINT 2008,KMIN
136      R=KMIN
137      T=KMIN
138      GOTO 100
139      8      MP=MPI-MI
140      DO 5000 I=1,MP
141      IF(INDEX(I).EQ.KMIN) GOTO 5002
142 5000      CONTINUE
143 5002      IA=I
144      K=MP-IA
145      DO 5001 I=1,K
146      IA2=IA+I-1
147      IA3=IA+I
148      INDEX(IA2)=INDEX(IA3)
149 5001      CONTINUE
150      MI=MI+1

```

```

151 MP=MP-1
152 PRINT 2006, NUMIT
153 MAUX=MP/10
154 MA=MP-10*MAUX
155 FMT4(7)=15*16 + 15*16*13
156 FMT4(7)=FMT4(7)+256*MAUX+MA
157 WRITE(6, FMT4)(INDEX(K), K=1, MP)
158 PRINT 2009, KMIN
159 R=KMIN
160 T=KMIN
161 GOTO 100
162 END

```

```

1 SUBROUTINE AMIN(KMIN)
2
3
4 C CETTE SOUS-ROUTINE RECHERCHE LA COMPOSANTE LA PLUS NEGATIVE DU
5 C POINT COMPLEMENTAIRE
6
7
8 IMPLICIT REAL *8(A-B, P)
9 IMPLICIT INTEGER(Q-Z)
10 COMMON AUX(10, 11), N, N1
11 VMIN=0.
12 KMIN=0
13 DO 1 J=1, N
14 IF(AUX(J, N1) .GE. VMIN) GOTO 1
15 VMIN=AUX(J, N1)
16 KMIN=J
17 1 CONTINUE
18 RETURN
19 END

```

```

1 SUBROUTINE SIGNE(SIGN, CAR)
2
3
4 C CETTE SOUS-ROUTINE DONNE LA CONFIGURATION DE SIGNES DU POINT
5 C COMPLEMENTAIRE
6
7
8 IMPLICIT INTEGER(Q-Z)
9 IMPLICIT REAL *8(A-B, P)
10 LOGICAL *1 SIGN(10), CAR(2)
11 COMMON AUX(10, 11), N, N1
12 DO 1 I=1, N
13 IF(AUX(I, N1)) 2, 3, 3
14 2 SIGN(I)=CAR(2)
15 GOTO 1
16 3 SIGN(I)=CAR(1)
17 1 CONTINUE
18 RETURN
19 END

```

C CETTE SOUS-ROUTINE VERIFIE LA POSITIVITE DES COMPOSANTES DU
C POINT COMPLEMENTAIRE

```

IMPLICIT INTEGER(Q-Z)
IMPLICIT REAL *8(A-B,P)
INTEGER *2 FMT3(14)
COMMON AUX(10,11),N,N1
DATA FMT3/'(1H+,45X,2X,00(F6.2,2X),1H))' /
DO 1 I=1,N
IF(AUX(I,N1).LE.0.) GOTO 2
1 CONTINUE
FMT3(7)=15#16+15#16#*3
FMT3(7)=FMT3(7)+256#NAUX+NA
PRINT 2010
2010 FORMAT(///,1X,'LE PT. DE COMPL. SOLUTION DU PROBLEME EST : (')
WRITE(6,FMT3)(AUX(I,N1),I=1,N)
STOP
2 DO 3 J=I,N
IF(DABS(AUX(J,N1)).LT.1.E-6) GOTO 4
3 CONTINUE
RETURN
4 PRINT 2001
2001 FORMAT(///,1X,'LE PROBLEME EST DEGENERE')
STOP
END

```

SUBROUTINE GENER

C CETTE SOUS-ROUTINE TRANSFORME UNE MATRICE GENEREE ALEATOIREMENT EN
C UNE MATRICE DIAGONALEMENT DOMINANTE

```

IMPLICIT REAL *8(A-B,P)
COMMON AUX(10,11),N,N1
READ 2008,IX
2008 FORMAT(I6)
DO 2000 I=1,N
DO 2001 J=1,N1
CALL RANDOU(IX,YFL)
AUX(I,J)=YFL
2001 CONTINUE
2000 CONTINUE
PRINT 2006
2006 FORMAT(1H1,////,4X,'LA MATRICE GENEREE ALEATOIREMENT EST :',///)
DO 2002 I=1,N
2002 PRINT 3000,(AUX(I,J),J=1,N1)
3000 FORMAT(1H0,11(1X,F6.2,1X))
DO 2003 I=1,N
B=0.
DO 2004 J=1,N
2004 B=B+DABS(AUX(I,J))
2003 AUX(I,I)=B
PRINT 2007
2007 FORMAT(////,1X,'LA MATRICE ALEATOIRE DIAG. DOMINANTE EST :',///)
DO 2005 I=1,N
2005 PRINT 3000,(AUX(I,J),J=1,N1)
RETURN
END

```

```

1      SUBROUTINE RANDOU(IX,YFL)
2
3
4 C      CETTE SOUS-ROUTINE GENERE UNE SEQUENCE ALEATOIRE DE NOMBRES
5
6
7      IMPLICIT REAL *8(A-B,P)
8      COMMON AUX(10,11),N,N1
9      IX=IX*65539
10     IF(IX.GE.0) GOTO 6
11     IX=IX+2147483647+1
12 6    YFL=DFLOAT(IX)
13     YFL=YFL*.4656613E-9
14     YFL=YFL*20-10
15     RETURN
16     END

1      SUBROUTINE PIVOT(R,T,NUMIT,Z,W)
2
3
4 C      CETTE SOUS-ROUTINE EFFECTUE UN PIVOTAGE
5
6
7      IMPLICIT INTEGER(G-Z)
8      IMPLICIT REAL *8(A-B,P)
9      DIMENSION Z(11),W(10)
10     COMMON AUX(10,11),N,N1
11     PIV=AUX(R,T)
12     AUX(R,T)=1/PIV
13     DO 1 J=1,N1
14     IF (J.EQ.T) GOTO 1
15     AUX(R,J)=-AUX(R,J)/PIV
16     DO 4 I=1,N
17     IF(I.EQ.R) GOTO 4
18     AUX(I,J)=AUX(I,J)+AUX(R,J)*AUX(I,T)
19 4    CONTINUE
20 1    CONTINUE
21     DO 2 I=1,N
22     IF(I.EQ.R) GOTO 2
23     AUX(I,T)=AUX(I,T)/PIV
24 2    CONTINUE
25     WZ=Z(T)
26     Z(T)=W(R)
27     W(R)=WZ
28     NUMIT=NUMIT+1
29     PRINT 2002,NUMIT
30 2002 FORMAT(1H1,1X,'LE TABLEAU DE L''ITERATION',I2,' EST :')
31     PRINT 2003,(Z(I),I=1,N1)
32 2003 FORMAT(///,5X,11(A6,2X))
33     DO 3 K=1,N
34 3    PRINT 2005,W(K),(AUX(K,I),I=1,N1)
35 2005 FORMAT(1H0,A4,11(1X,F6.2,1X))
36     RETURN
37     END

```


B Algorithme de Murty

Cet algorithme ne différant de celui de Bard que par une seule sous-routine, nous ne présenterons ici que cette sous-routine.

SUBROUTINE AMIN(KMIN)

CETTE SOUS-ROUTINE RECHERCHE LA PREMIERE COMPOSANTE NEGATIVE DU POINT COMPLEMENTAIRE

IMPLICIT INTEGER(Q-Z)
 IMPLICIT REAL*8(A-B,P)
 REAL*8 VMIN
 COMMON AUX(10,11),N,N1
 DO 1 J=1,N
 IF (AUX(J,N1) .GE. 0.) GOTO 1
 KMIN=J
 GOTO 2
 CONTINUE
 RETURN
 END

C Algorithme de Kostreva

PROGRAM KOSTR

C TABLE DES VARIABLES

C *****

C A : MATRICE NXN DES DONNEES

C AUX : MATRICE TRANSITOIRE

C Q : VECTEUR POINT COMPLEMENTAIRE

C INDEX : ENSEMBLE D'INDEXATION

C N : DIMENSION DE LA MATRICE DES DONNEES

C N1 : DIMENSION TOTALE DES DONNEES

C P : INDICE DE LA LIGNE DU PIVOT

C T : INDICE DE LA COLONNE DU PIVOT

C NUMIT : COMPTEUR DU NOMBRE D'ITERATIONS

C PIV : COMPTEUR DU NOMBRE DE PIVOTAGES

C NBRE : VECTEUR DESIGNANT TOUTES LES PARTIES DE

C L'ENSEMBLE <1,2,3,...,2**N>

C SUB : VECTEUR DE CONFIGURATION DE SIGNES ASSOCIE AU VECTEUR BIN

C BIN : VECTEUR DE DECOMPOSITION BINAIRE D'UN NOMBRE

C SIGN : VECTEUR LOGIQUE DE LA CONFIGURATION DE SIGNES DU

C POINT COMPLEMENTAIRE

C CAR : VECTEUR LOGIQUE COMPOSE DES DEUX VARIABLES + ET -

C TOUTES LES AUTRES VARIABLES SONT DES VARIABLES INTERMEDIAIRES

IMPLICIT REAL *8(A-B)

IMPLICIT INTEGER (Q-Z)

LOGICAL SIGN*1(10),CAR*1(2),SE*1(10)

INTEGER FMT1*2(14),FMT2*2(13),FMT4*2(13)

INTEGER SUBSIG*2(27),SUBSI*2(21)

INTEGER BIN,P,PU,PUR

DIMENSION Z(11),W(10),ZU(11),WU(10),INDEX(10),BIN(11),SUB(1024,10)

C,NBRE(1024)

COMMON AUX(10,11),N,N1

DIMENSION A(1,11)

DATA GAUX/' Q' /

DATA SUBSIG/' (1X,12,3X,1H<,00(I3,2H),1H>,00X,1H(,00(A1,3X),1H), /

C)' /

DATA SUBSI/' (1X,12,3X,5H< >,65X,1H(,00(A1,3X),1H), /)' /

DATA FMT1/' (1H+,10X,3X,00(F6.2,3X),1H)' /

DATA FMT2/' (1H+,10X,3X,00(A1,3X),1H)' /

DATA FMT4/' (1H+,10X,3X,00(I2,3X),1H>)' /

READ 1000,N

FORMAT(I2)

C INITIALISATION

```

51
52
53 NUMIT=0
54 N1=N+1
55 NAUX=N/10
56 NA=N-10*NAUX
57
58

```

```

59 C CONSTRUCTION DU TABLEAU DES BIJECTIONS
60
61

```

```

62 CALL GENER(A)
63 READ 1002,(ZU(I),I=1,N)
64 READ 1002,(WU(I),I=1,N)
65 1002 FORMAT(10A4)
66 DO 10 I=1,N
67 Z(I)=ZU(I)
68 10 W(I)=WU(I)
69 READ 1003,(CAR(I),I=1,2)
70 1003 FORMAT(2A1)
71 Z(N1)=QAUX
72 DO 2 I=1,N
73 AUX(I,N1)=A(I,N1)
74 DO 2 J=1,N
75 2 AUX(I,J)=A(I,J)
76 PRINT 2004
77 2004 FORMAT(1H1,////,12X,'METHODE DE KOSTREVA',////)
78 SUBSIG(20)=15*16+15*16*3
79 SUBSIG(20)=SUBSIG(20)+256*NAUX+NA
80 SUBSI(14)=15*16+15*16*3
81 SUBSI(14)=SUBSI(14)+256*NAUX+NA
82 N2=2*N
83 DO 200 I=1,N2
84 IM1=I-1
85 CALL BINARY(IM1,BIN)
86 NBRE(I)=BIN(N+1)
87 CALL SIG(SE,BIN,CAR)
88 CALL SUBENS(I,BIN,SUB)
89 SUBSIG(16)=15*16+15*16*3
90 INT=68-5*BIN(N+1)
91 NAUX=INT/10
92 MA=INT-10*NAUX
93 SUBSIG(16)=SUBSIG(16)+256*MAUX+MA
94 SUBSIG(8)=15*16+15*16*3
95 IN=I*IN(N+1)
96 LAUX=IN/10
97 LA=IN-10*LAUX
98 SUBSIG(8)=SUBSIG(8)+256*LAUX+LA
99 LI=BIN(N+1)
100 IF(LI.NE.0) GOTO 3000

```

```

WRITE(6, SUBSI)(IM1, (SE(K), K=1, N))
GOTO 200
3000 WRITE(6, SUBSIG)(IM1, (SUB(I, K), K=1, LI), (SE(K), K=1, N))
200  CONTINUE

C      PREMIERE ETAPE DE L'ALGORITHME :
C      ON RECHERCHE - LE POINT COMPLEMENTAIRE
C      - SA CONFIGURATION DE SIGNES
C      ON VERIFIE SA POSITIVITE
C      ON DEDUIT L'ENSEMBLE D'INDEXATION QUI NOUS PERMET
C      DE TROUVER L'ELEMENT PIVOT

PUR=0
CALL PERMUT(PUR)
PUR=PUR+1
PRINT 2006, NUMIT
2006  FORMAT(///, 1X, 'POUR L''ITERATION SUIVANTE, L''ENSEMBLE'
C, ' D''INDEXATION NUMERO ', I2, ' EST : ', ///, 10X, '<')
IF(PUR.EQ.1) GOTO 7000
LU=NBRE(PUR)
J AUX=LU/10
JA=LU-10*JAUX
FMT4(7)=15*16+15*16**3
FMT4(7)=FMT4(7)+256*JAUX+JA
WRITE(6, FMT4)(SUB(PUR, K), K=1, LU)
NUMIT=NUMIT+1
PRINT 2008, NUMIT
2008  FORMAT(1H1, //, 1X, 'LE(LES) TABLEAU(X) DE L''ITERATION', I3, ' EST(
CSONT) :')
DO 101 I=1, LU
T=SUB(PUR, I)
R=SUB(PUR, I)
CALL PIVOT(R, T, Z, W)
101  CONTINUE
500  FMT1(7)=15*16+15*16**3

C      IEME ETAPE DE L'ALGORITHME
C      ON EFFECTUE UN PIVOTAGE
C      ON RECHERCHE - LE POINT COMPLEMENTAIRE
C      - SA CONFIGURATION DE SIGNES
C      ON VERIFIE SA POSITIVITE
C      ON DEDUIT L'ENSEMBLE D'INDEXATION SUIVANT

FMT2(7)=15*16+15*16**3
FMT1(7)=FMT1(7)+256*NAUX+NA
PRINT 2000

```

```

51 2000 FORMAT(///,1X,'LE POINT DE COMPLEMENTARITE EST :',///,10X,'(')
52 WRITE(6,FMT1)(AUX(I,N1),I=1,N)
53 CALL SIGNE(SIGN,CAR)
54 FMT2(7)=FMT2(7)+256*NAUX+NA
55 PRINT 2001
56 2001 FORMAT(///,1X,'LA CONFIGURATION DE SIGNES DU POINT '
57 C'DE COMPLEMENTARITE EST :',///,10X,'(')
58 WRITE(6,FMT2)(SIGN(I),I=1,N)
59 CALL POSIT (NA,NAUX)
60 PU=0
61 DO 100 I=1,N
62 P=0
63 IF(AUX(I,N1).LT.0.) P=1
64 PU=PU+P*2**(N-I)
65 100 CONTINUE
66 CALL PERMUT(PU)
67 PU=PU+1
68 PRINT 2006,NUMIT
69 NUMIT=NUMIT+1
70 IF(PU.EQ.1) GOTO 7000
71 FMT4(7)=15*16+15*16**3
72 LU=NBRE(PU)
73 JAUX=LU/10
74 JA=LU-10*JAUX
75 FMT4(7)=FMT4(7)+256*JAUX+JA
76 WRITE(6,FMT4)(SUB(PU,K),K=1,LU)
77 DO 8000 I=1,N1
78 AUX(I,N1)=A(I,N1)
79 DO 8000 J=1,N
80 8000 AUX(I,J)=A(I,J)
81 DO 11 I=1,N
82 Z(I)=ZU(I)
83 11 W(I)=WU(I)
84 PRINT 2008,NUMIT
85 DO 102 I=1,LU
86 R=SUB(PU,I)
87 T=SUB(PU,I)
88 CALL PIVOT(R,T,Z,W)
89 102 CONTINUE
90 GOTO 500
91 7000 PRINT 7001
92 7001 FORMAT(1H+,10X,3X,'>')
93 PRINT 2008,NUMIT
94 PRINT 9001,(Z(I),I=1,N1)
95 9000 FORMAT(///,5X,11(A6,2X))
96 DO 9001 K=1,N
97 9001 PRINT 9002,W(K),(A(K,I),I=1,N1)
98 9002 FORMAT(1H0,A4,11(1X,F6.2,1X))
99 DO 9003 I=1,N
100 9003 AUX(I,N1)=A(I,N1)
101 GOTO 500
102 END

```

```

1      SUBROUTINE PIVOT(R,T,7,W)
2
3
4 C      CETTE SOUS-ROUTINE EFFECTUE UN PIVOTAGE
5
6
7      IMPLICIT REAL *8(A,P)
8      IMPLICIT INTEGER(R-Z)
9      DIMENSION Z(11),W(10)
10     COMMON AUX(10,11),N,N1
11     PIV=AUX(R,T)
12     AUX(R,T)=1/PIV
13     DO 1 J=1,N1
14     IF(J.EQ.T) GOTO 1
15     AUX(P,J)=-AUX(R,J)/PIV
16     DO 4 I=1,N
17     IF(I.EQ.R) GOTO 4
18     AUX(I,J)=AUX(I,J)+AUX(R,J)*AUX(I,T)
19 4    CONTINUE
20 1    CONTINUE
21     DO 2 I=1,N
22     IF(I.EQ.R) GOTO 2
23     AUX(I,T)=AUX(I,T)/PIV
24 2    CONTINUE
25     WZ=Z(T)
26     Z(T)=W(R)
27     W(R)=WZ
28     PRINT 2003,(Z(I),I=1,N1)
29 2003 FORMAT(///,5X,11(A6,2X))
30     DO 3 K=1,N
31 3    PRINT 2005,W(K),(AUX(K,I),I=1,N1)
32 2005 FORMAT(1H0,A4,11(1X,F6.2,1X))
33     RETURN
34     END

```

```

1      SUBROUTINE RANDOU(IX,YFL)
2
3
4 C      CETTE SOUS-ROUTINE GENERE UNE SEQUENCE ALEATOIRE DE NOMBRES
5
6
7      IMPLICIT REAL *8(A,Y)
8      COMMON AUX(10,11),N,N1
9      IX=IX#65539
10     IF(IX.GE.0) GOTO 6
11     IX=IX+2147483647+1
12 6    YFL=DFLOAT(IX)
13     YFL=YFL#.4656613E-9
14     YFL=YFL#20-10
15     RETURN
16     END

```

SUBROUTINE GENER(A)

CETTE SOUS-ROUTINE TRANSFORME UNE MATRICE GENEREE ALEATOIREMENT EN
UNE MATRICE DIAGONALEMENT DOMINANTE

```

C
C
IMPLICIT REAL *8(A,B,Y)
DIMENSION A(10,11)
COMMON AUX(10,11),N,N1
2008 READ 2008,IX
      FORMAT(I6)
      DO 2000 I=1,N
      DO 2001 J=1,N1
      CALL RANDOU(IX,YFL)
2001 A(I,J)=YFL
2000 CONTINUE
      PRINT 2006
2006 FORMAT(1H1,////,4X,'LA MATRICE GENEREE ALEATOIREMENT EST :',//)
      DO 2002 I=1,N
2002 PRINT 3000,(A(I,J),J=1,N1)
3000 FORMAT(1H0,11(1X,F6.2,1X))
      DO 2003 I=1,N
      B=0.
      DO 2004 J=1,N
2004 B=B+DABS(A(I,J))
2003 A(I,I)=B
      PRINT 2007
2007 FORMAT(////,1X,'LA MATRICE ALEATOIRE DIAG. DOMINANTE EST :',//)
2005 DO 2005 I=1,N
      PRINT 3000,(A(I,J),J=1,N1)
      RETURN
      END

```

SUBROUTINE SUBENS(I,BIN,SUB)

CETTE SOUS-ROUTINE CALCULE UN SOUS ENSEMBLE A PARTIR DE LA
DECOMPOSITION BINAIRE D'UN NOMBRE

```

C
C
IMPLICIT REAL *8(A)
INTEGER BIN(11),SUB(1024,10)
COMMON AUX(10,11),N,N1
J=1
DO 501 K=1,N
IF(BIN(K).EQ.0) GOTO 501
SUB(I,J)=K
J=J+1
501 CONTINUE
      RETURN
      END

```

1 SUBROUTINE POSIT(NA,NAUX)

2
3
4 C CETTE SOUS-ROUTINE VERIFIE LA POSITIVITE DES COMPOSANTES DU
5 C POINT COMPLEMENTAIRE

6
7
8 IMPLICIT REAL *8(A)

9 INTEGER *2 FMT3(14)

10 COMMON AUX(10,11),N,N1

11 DATA FMT3/'(1H+,45X,2X,00(F6.2,2X),1H))'/'

12 DO 1 I=1,N

13 IF(AUX(I,N1).LE.0.) GOTO2

14 1 CONTINUE

15 FMT3(7)=15*16+15*16**3

16 FMT3(7)=FMT3(7)+256*NAUX+NA

17 PRINT 2000

18 2000 FORMAT(///,1X,'LE PT. DE COMPL., SOLUTION DU PROBLEME EST : (')

19 WRITE(6,FMT3)(AUX(I,N1),I=1,N)

20 STOP

21 2 DO 3 J=I,N

22 IF(DABS(AUX(J,N1)).LT.1.E-6) GOTO 4

23 3 CONTINUE

24 RETURN

25 4 PRINT 2001

26 2001 FORMAT(///,1X,'LE PROBLEME EST DEGENERE')

27 STOP

28 END

1 SUBROUTINE BINARY(P,BIN)

2
3
4 C CETTE SOUS-ROUTINE DONNE LA DECOMPOSITION BINAIRE D'UN NOMBRE

5
6
7 IMPLICIT REAL *8(A)

8 IMPLICIT INTEGER(P)

9 INTEGER BIN(11)

10 COMMON AUX(10,11),N,N1

11 PE=P

12 BIN(N+1)=0

13 DO 301 I=1,N

14 J=N-I+1

15 BIN(J)=0

16 PDEMI=PE/(2**I)

17 IF(PE.EQ.(2**I)*PDEMI) GOTO 302

18 BIN(J)=1

19 BIN(N+1)=BIN(N+1)+1

20 302 PE=PE-BIN(J)*2**(I-1)

21 301 CONTINUE

22 RETURN

23 END


```

1 SUBROUTINE PERMUT(P)
2
3
4 C     CETTE SOUS-ROUTINE EFFECTUE UNE PERMUTATION D'INDICES
5
6
7 IMPLICIT REAL *8(A)
8 INTEGER P
9 COMMON AUX(10,11),N,N1
0 RETURN
1 END

```

```

1 SUBROUTINE SIGNE(SIGN,CAR)
2
3
4 C     CETTE SOUS-ROUTINE DONNE LA CONFIGURATION DE SIGNES DU POINT
5 C     COMPLEMENTAIRE
6
7
8 IMPLICIT REAL *8(A)
9 LOGICAL *1 SIGN(10),CAR(2)
0 COMMON AUX(10,11),N,N1
1   DO 1 I=1,N
2     IF(AUX(I,N1)) 2,3,3
3 2   SIGN(I)=CAR(2)
4     GOTO 1
5 3   SIGN(I)=CAR(1)
6 1   CONTINUE
7   RETURN
8   END

```

```

1 SUBROUTINE SIG(SE,BIN,CAR)
2
3
4 C     CETTE SOUS-ROUTINE DONNE LA CONFIGURATION DE SIGNES ASSOCIEE A
5 C     UN NOMBRE
6 C     ELLE ASSOCIE AU CHIFFRE 1 LE SIGNE +
7 C     AU CHIFFRE 0 LE SIGNE -
8
9
0 IMPLICIT REAL *8(A)
1 LOGICAL *1 CAR(2),SE(10)
2 INTEGER BIN(11)
3 COMMON AUX(10,11),N,N1
4 DO 400 I=1,N
5   SE(I)=CAR(2)
6   IF(BIN(I).EQ.0) SE(I)=CAR(1)
7 400 CONTINUE
8   RETURN
9   END

```

TABLEAU DE COMPARAISON DU NOMBRE DE PIVOTAGES

Avant de passer aux résultats, signalons que les matrices de données ont été générées chacune par une séquence aléatoire de nombres, à partir d'un nombre lu.

Exemple	Bard	Murty	Kostreva (identité)	Kostreva (permutation)
1	4	4	10	12
2	8	10	16	18
3	2	2	3	3
4	6	8	32	45
5	4	6	10	13
6	4	4	6	7
7	6	8	14	18
8	5	5	5	6
9	8	10	18	24
10	7	13	64	35
11	8	8	8	8
12	3	5	3	5
13	4	4	4	6
14	8	8	7	11
15	4	4	8	10
16	6	6	12	14
17	5	5	5	6
18	6	6	6	8
19	6	8	32	19
20	4	4	8	10

μ Bard = 5,4
 μ Murty = 6,4
 μ Kostreva(id.) = 13,55
 μ Kostreva(perm.) = 13,9

D'après le tableau ci-dessus, l'algorithme de Bard résoud plus rapidement les problèmes linéaires de complémentarité que l'algorithme de Murty. Cette conclusion nécessite cependant une étude statistique plus fouillée. Cependant, si nous observons de plus près le schéma de ces deux algorithmes nous constatons qu'ils diffèrent uniquement entre eux par un seul test de minimisation sur les composantes du point complémentaire. Pour l'algorithme de Murty, ce test recherche la première composante négative du point complémentaire tandis que, pour l'algorithme de Bard, il recherche la composante la plus négative. Ceci justifierait la moyenne moins élevée du nombre de pivotages.

4 COMPARAISON DES ALGORITHMES DE MURTY ET DE KOSTREVA
 =====
 MUNI DE L'IDENTITE
 =====

Avant d'aborder ce sujet, il nous paraît important de signaler le fait suivant : Dans l'algorithme de Kostreva, à chaque itération, l'opération pivotage s'effectue directement sur le tableau initial. Le choix est apparemment plus avantageux que d'effectuer l'opération pivotage à partir du dernier tableau de l'itération précédente (un échantillonnage plus grand permettrait peut-être d'autres conclusions). Avant de passer à la comparaison numérique, analysons le test sur lequel on s'appuie. Il s'agit du test du signe [10]. Signalons tout d'abord que la théorie des tests sert à rejeter ou à ne pas rejeter une hypothèse. Le problème consiste donc à tester une hypothèse H_0 contre une alternative H_1 . Ce test du signe est un test basé sur une technique non paramétrique. Il ne demande aucune hypothèse de départ et, comme son nom l'indique, il se base essentiellement sur la configuration de signe de la différence des éléments des paires des différentes populations. Dans la suite, nous noterons par R le signe de la différence entre les valeurs des résultats obtenus pour un même exemple testé.

A Etude du test du signe

Hypothèse : Soient - A et B deux populations

- l'hypothèse nulle H_0 : les éléments de la population A égalent les éléments de la population B
- l'hypothèse alternative H_1 : les éléments de la population A sont supérieurs aux éléments de la population B

- R : la variable aléatoire qui dénombre les signes positifs dans les différences des éléments des paires des deux populations.

Thèse : Déterminer le seuil critique R^* de la région de rejet de l'hypothèse nulle.

Résolution du problème :

Distribution d'échantillonnage de la statistique R :

On est en présence de n paires d'éléments (en éliminant les "ties")

- . d'une probabilité $p = 1/2$ si l'hypothèse H_0 est vérifiée.

C'est pourquoi, on va appliquer un modèle binomial : $Bi(n, 1/2)$

Remarque : les tables de la distribution binomiale cumulée donneront les probabilités $p(R \leq r)$.

Règle de décision :

Soient R^* le seuil critique de la région de rejet de l'hypothèse H_0

- . $\alpha = 0,05$ (par exemple) le niveau de signification du test.

On choisit R^* tel que $P_{H_0}(R > R^*) \leq \alpha$

Si le R observé est supérieur à R^* , on rejette l'hypothèse nulle au niveau d'incertitude $\alpha = 0.05$ c'est-à-dire, ou bien l'hypothèse H_0 est fautive, ou bien un événement de faible probabilité s'est produit.

B Autre possibilité

On aurait pu considérer :

H_0 : les éléments de la population A égalent les éléments de la population B.

H_1 : les éléments de la population B sont supérieurs aux éléments de la population A.

Dans ce cas, on détermine R^* tel que $P_{H_0} (R < R^*) \leq \alpha$

Et 1°) on rejette l'hypothèse H_0 si $R < R^*$

2°) on ne rejette pas l'hypothèse H_0 si $R \geq R^*$

C Le cas de Murty et Kostreva

Soient A (= l'algorithme de Kostreva) et B (= l'algorithme de Murty), les deux algorithmes en compétition.

Nous désirons comparer la rapidité d'exécution (du point de vue du nombre de pivotages) de ces deux algorithmes.

C'est pourquoi, nous prenons comme :

1°) hypothèse nulle H_0 : $A = B$: les deux algorithmes effectuent le même nombre de pivotages;

2°) hypothèse alternative H_1 : $A > B$: l'algorithme de Murty possède un nombre de pivotages inférieur à celui de Kostreva.

Reprenons le tableau du nombre de pivotages. Nous mettrons sous la dénomination : "Ties" une égalité entre les nombres d'itérations pour un même exemple testé.

Kostreva	Murty	différence	signe
10	4	+6	+
16	10	+6	+
3	2	+1	+
32	8	+24	+
10	6	+4	+
6	4	+2	+
14	8	+6	+
5	5	0	"ties"
18	10	+8	+
64	13	+51	+
8	8	0	"ties"
3	5	-2	-
4	4	0	"ties"
7	8	-1	-
8	4	+4	+
12	6	+6	+
5	5	0	"ties"
6	6	0	"ties"
32	8	+24	+
8	4	+4	+

C'est pourquoi, nous avons :

$$m = 20 - 5 = 15$$

$$h = 1/2$$

$$R = 13$$

De plus, les tables de la distribution binômiale cumulée nous donnent :

$$\begin{aligned} P(R > 10) &= 1 - P(R \leq 10) \\ &= 1 - 0.94077 \\ &= 0.03923 \end{aligned}$$

$$\begin{aligned} P(R > 9) &= 1 - P(R \leq 9) \\ &= 1 - 0.84912 \\ &= 0.15088 \end{aligned}$$

Donc, d'après la règle de décision, nous avons :

$$R^* = 10$$

Nous allons donc . rejeter l'hypothèse H_0 si $R > 10$,
 . ne pas rejeter l'hypothèse H_0 si $R \leq 10$
 L'hypothèse H_0 est donc rejetée (car $R = 13$) au niveau d'incertitude 0.05.

D Explication de ce résultat

Ce résultat n'est pas étonnant car, comme dans l'algorithme de Kostreva, l'élément pivot est déterminé à partir des données de bijections et d'une permutation, nous pouvons nous trouver en présence de pivotages "inutiles".

E Remarque

Une étude analogue aurait aussi bien pu se faire entre les algorithmes de Bard et de Kostreva, ou encore entre Bard et Kostreva muni d'une permutation distincte de l'identité.

5 COMPARAISON DE L'ALGORITHME DE KOSTREVA MUNI DE
L'IDENTITE ET MUNI D'UNE PERMUTATION QUELCONQUE

La permutation quelconque que nous avons considérée est celle qui envoie chaque élément sur son suivant et le dernier sur le premier.

$$\sigma: A \longrightarrow A \quad \text{où } A = \{1, 2, \dots, 2^m\}$$

$$i \rightsquigarrow \begin{cases} i+1 & \text{si } i \neq 2^m \\ 1 & \text{sinon} \end{cases}$$

Les résultats obtenus dans le tableau du nombre de pivotages nous convient à travailler, à l'avenir, plutôt avec la permutation identité.

Une étude analogue à celle du point précédent peut se faire sans aucune difficulté. Les résultats obtenus sont :

$$n = 20 - 2 = 18$$

$$p = 1/2$$

$$R = 15$$

$$\alpha = 0.05$$

$$R^* = 6 \quad \text{car } P(R > 6) = 1 - P(R \leq 6) = 1 - 0.9519$$

$$= 0.0481$$

$$P(R > 7) = 1 - P(R \leq 7) = 1 - 0.8811$$

$$= 0.1189$$

Nous rejetterons l'hypothèse : "l'algorithme de Kostreva muni d'une permutation quelconque effectue le même nombre de pivotages que celui de Kostreva muni de l'identité" si $R > R^*$, ce qui est le cas puisque $R = 15$.

Les conclusions que nous avons tirées dans ce chapitre à propos des problèmes linéaires de complémentarité, sont à prendre en considération malgré le petit nombre d'exemples testés. En effet, l'important n'est pas le nombre d'exemples considérés, mais l'utilisation de "données" représentatives de la population. Dans notre cas, nous avons traité pour chacun des algorithmes les mêmes vingt problèmes linéaires dont les données sont générées aléatoirement.

C O N C L U S I O N

Dans ce mémoire, nous avons rencontré deux types d'algorithmes directs, les algorithmes directs de type pivot unique illustrés par les schémas de Bard et de Murty et les algorithmes directs de type pivot bloc illustrés par l'algorithme de Kostreva.

La programmation des schémas de Bard et Murty nous a permis de constater que l'algorithme de Bard est le plus efficace en temps d'exécution.

Quant à l'algorithme de Kostreva, sur la base d'un test statistique : le test du signe, nous l'avons comparé aux deux précédents. Cette comparaison basée sur le nombre de pivotages nous a amené à conclure que ce dernier algorithme est le moins efficace des trois quant au temps d'exécution . Cette conclusion est en opposition avec celle de Kostreva basée sur le nombre d'itérations.

B I B L I O G R A P H I E

= = = = =

1. Brown, K.M., "Computer Oriented Algorithms for Solving Systems of Simultaneous Nonlinear Algebraic Equations", Numerical Solution of Systems of Nonlinear Algebraic Equations, Ed. : G.D. Byrne and C.A. Hall, Academic Press, New York, 1973.
2. Chandrasekaran, R., "A special Case of the Complementary Pivot Problem", Opsearch, 7, 1970, 263-268.
3. Cottle, R., "Nonlinear Programs with Positively Bounded Jacobians", SIAM J. Appl. Math., 14, 1, 1966, 147-158.
4. Cottle, R., "Numerical Methods for Complementarity problems in Engineering and Applied Science", in Troisième Colloque International sur les Méthodes de Calcul Scientifique et Technique, Versailles-Rocquencourt, 5-9 décembre 1977.
5. Cottle, R., and Dantzig, G., "Complementary Pivot Theory", Mathematics of the Decision Science Part 1, Ed. : G.B. Dantzig and A.F. Veinott, Jr., American Mathematical Society, Providence, 1968. (Also Lin. Alg. Appl. 1, 1968, 103-125 and Studies in Optimization, Ed. G.B. Dantzig and B.C. Eaves, MAA, 1974).
6. Cottle, R., and Dantzig, G., "Positive (semi) Definite Programming" in Nonlinear Programming, Ed. J. Abadie, North-Holland, Amsterdam, 1967, 55-73.
7. Dantzig, G., Linear Programming and Extensions, Princeton Univ. Press, Princeton, New Jersey, 1963.
8. Eaves, B., "On the Basic Theorem of Complementarity", Math. Prog 1, 1, 1971, 68-75.
9. Fiedler, M. and Ptak, V., "On Matrices with Non-positive Off-diagonal Elements and Positive Principal Minors", Czech. Math. Journal, 12, 1962, 382-400.

10. Cuenther, W., Concepts of Statistical Inference, International Student Edition, Mc. Graw-Hill, Kogakusha.
11. Karamardian, S., "The Complementarity Problem", Math. Prog., 2, 1, 1972, 107-129.
12. Kostreva, M., "Direct Algorithms for Complementarity Problems", Ph. D. Thesis, Rensselaer Polytechnic Institute, Troy, N.Y., 1976.
13. Kuhn, H. and Tucker, A., (Nonlinear Programming) in Second Berkeley Symposium on Mathematical Statistics and Probability, Ed. J. Neyman, Univ. of Calif. Press., Berkeley, Calif., 1951.
14. Lemke, C., "Bimatrix Equilibrium Points and Mathematical Programming", Management Science, 11, 1965, 681-689.
15. Lemke, C., "On Complementarity Pivot Theory", Mathematics of the Decision Science Part I, Ed. G.B. Dantzig and A.F. Veinott, Jr., American Mathematical Society, Providence, 1968.
16. Lemke, C., "Recent Results on Complementarity Problems", Nonlinear Programming, Ed. J. B. Rosen, O.L. Magasarian, and K. Ritter, Academic Press, N.Y., 1970.
17. Moré, J., "Coercivity Conditions in Nonlinear Complementarity Problems", SIAM Review, 16, 1, 1974, 1-15.
18. Moré, J., and Rheinboldt, W., "On P- and S-Functions and Related Classes of n-dimensional Nonlinear Mappings". Lin. Alg. and Appl., 6, 1973, 45-68.
19. Murty, K., "On the Number of Solutions to the Complementarity Problem and Spanning Properties of Complementary Cones", Lin. Alg. and Appl., 5, 1972, 65-108.

20. Murty, K., "Note on a Bard-Type Schema For Solving the Complementarity Problem", *Opsearch*, 11, 2-3, 1974, 121-130.
21. von Neumann, J.; "Discussion of a Maximum Problem", Collected works VI (A.H. Taub, Ed.), Pergamon, N.Y., 1963.
22. Samelson, H., Thrall, R., and Wesler, O., "A partition Theorem for Euclidean n-Space", *AMS Proceedings*, 9, 1958, 805-807.
23. Strang, G., "Some Recent Contributions to Plasticity Theory", *J. Franklin Institute*, 302, 1976, 429-441.

T A B L E D E S M A T I E R E S

Introduction	4
Lexique	4
Chapitre I - <u>NOTIONS FONDAMENTALES</u>	
1 Position du problème	7
2 Interprétation géométrique des cônes complémentaires	15
3 Théorèmes d'existence et d'unicité	18
4 Quelques algorithmes disponibles	20
Algorithme de Lemke	20
Algorithme de Cottle et Dantzig	32
Chapitre II - <u>CARACTERISATION DE LA CLASSE P</u>	
Introduction	33
1 Définitions importantes	34
2 Position du problème	36
3 Définition et démonstration du caractère bijectif de S	37
A Définition de la correspondance S	37
B Le caractère surjectif de S	38
C La correspondance S est partout définie et fonctionnelle	39
D Le caractère injectif de l'application S	45
E Conclusions	46
4 Définition et démonstration du caractère surjectif de g	47
A Définition de l'application g	47
B Le caractère surjectif de l'application g	47
5 Résultat fondamental	49
6 La propriété du "trou du pigeon" pour les P-fonctions uniformes	51

7	La propriété du "trou du pigeon" dans le cas linéaire	54
	A Lien avec les P-matrices	54
	B Caractérisation des ensembles A_M et H_M	55
	C Une caractérisation des P-matrices : la propriété du "trou du pigeon"	59

Chapitre III - LES ALGORITHMES DIRECTS

	Introduction	61
1	Les algorithmes directs	62
	A Définitions	62
	B Organigramme général d'un algorithme direct	64
	C Remarque	65
	D Les différents types d'algorithmes directs	65
2	Deux exemples d'algorithmes directs de pivot unique	67
	A Algorithme de Bard	67
	B Algorithme de Murty	77
3	Algorithme de Kostreva	83
	A Préliminaire portant sur la théorie des groupes	83
	B Les données de l'algorithme et leur forme	84
	C Description et organigramme	85
	D Convergence de l'algorithme	89
	E Exemples numériques	91
	F Exemple montrant la nécessité de l'hypothèse : "non-dégénérescence"	94
	G Remarques	95
	H Exemple programmé	96

Chapitre IV - COMPARAISON SUCCINCTE DES TROIS ALGORITHMES PRESENTES

	Introduction	99
1	Programmation	101
	A Algorithme de Bard	101

B	Algorithme de Murty	107
C	Algorithme de Kostreva	108
2	Tableau de comparaison du nombre de pivotages	116
3	Comparaison des algorithmes de Bard et de Murty	117
4	Comparaison des algorithmes de Murty et de Kostreva muni de l'identité	118
A	Etude du test du signe	118
B	Autre possibilité	120
C	Le cas de Murty et de Kostreva	120
D	Explication de ce résultat	122
E	Remarque	122
5	Comparaison de l'algorithme de Kostreva muni de l'identité et muni d'une permutation quelconque	123

Conclusion

Bibliographie

Table des matières