Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

THESIS / THÈSE

MASTER EN SCIENCES MATHÉMATIQUES

Une méthode d'optimisation par approximations convexes successives

ENCKELS, Martine; VAN VYVE, Philippe

Award date:
1984

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.

Download date: 02. May. 2026

ANNEE ACADEMIQUE 1983 - 1984

UNE METHODE D'OPTIMISATION

PAR APPROXIMATIONS CONVEXES SUCCESSIVES

Promoteur :
V.H. NGUYEN

Martine ENCKELS
Philippe VAN VYVE

Nous tenons à remercier Monsieur V.H. NGUYEN en tant que promoteur et pour le sujet intéressant qu'il nous a proposé.

Nous désirons remercier particulièrement Monsieur J.J. STRODIOT pour son aide efficace ainsi que toutes les personnes qui nous ont encouragés.

INTRODUCTION

L'étude de la minimisation sous contraintes de fonctions non linéaires trouve son application dans de multiples domaines tels que l'économie ou la physique. Pour ne citer qu'un exemple, supposons qu'il y ait un modèle mathématique d'un phénomène mesuré en laboratoire. A partir de ce modèle, la mesure f du phénomène est connue sous la forme $f(t) = x_1 + x_2 e^{-x_3 t}$. L'objectif du laboratoire est de trouver les paramètres inconnus x_1, x_2, x_3 en mesurant les valeurs de f aux temps t^1, t^2, ...t^M. Il est raisonnable de se demander quelles sont les valeurs de x_1, x_2, x_3 qui soient optimales au sens des moindres carrés. Ce qui nous donne le problème suivant :

$$\min \sum_{i=1}^{M} (f(t^i) - x_1 - x_2 - e^{-x_3 t^i})^2$$

$$\text{sous contraintes } x_3 \geq 0$$

$$x_1 + x_2^2 = 1$$

où les contraintes dépendent du modèle mathématique.

L'objet de ce mémoire consiste à traiter un problème non linéaire (PNL) qui se présente sous la forme générale suivante :

$$\min \quad f(x)$$
$$\text{s.c.} \quad g(x) \leq 0$$
$$h(x) = 0$$

où chaque fonction a la particularité d'être de classe C^2 et généralement Lipschitzienne jusqu'à l'ordre 2. Plus précisément, notre but est de déterminer un point de Karush-Kuhn-Tucker.

Au chapitre I, nous exposons brièvement quelques méthodes de linéarisation telles que la méthode des plans

sécants et la "reverse convex programming" présentant les schémas algorithmiques.

Cependant il est possible de traduire notre problème en termes de minimisation de la $\| \ \|$ sur les conditions Karush-Kuhn-Tucker. Vu le caractère non différentiable de la norme, ceci nous conduit au chapitre II à l'étude de la minimisation d'une classe de fonctions non différentiables telle que J. Kreuser l'a présentée dans sa thèse en 1979 à l'Université de Wisconsin-Madison.

Le principe de résolution est essentiellement basé sur des approximations convexes successives de la fonction non différentiable. Ces approximations font ensuite l'objet d'une minimisation supposée plus aisée à résoudre que le problème original, de sorte que la solution de la minimisation fournit une direction de descente pour la fonction non différentiable et permet ainsi d'entamer une recherche linéaire.

Poursuivant notre étude, au chapitre III, nous décrivons un algorithme général que nous particularisons au problème initial en utilisant la règle d'Armijo dans la recherche linéaire. Pour un choix particulier de la norme comme la $\| \ \|_1$ ou la $\| \ \|_\infty$, la minimisation de chaque approximation convexe donne lieu à un problème linéaire. D'où l'intérêt pour pouvoir appliquer le SIMPLEXE. Tel est l'objet du chapitre IV. Enfin, au chapitre V et en annexe, se trouvent l'implémentation de l'algorithme ainsi que les principaux résultats des exemples qui ont été testés.

TABLE DES MATIERES

INTRODUCTION

CHAPITRE I : DESCRIPTION DU PROBLEME.

1. Formulation du problème 1
2. Evocation de différentes méthodes . . . 2

CHAPITRE II : ETUDE DE LA MINIMISATION D'UNE CLASSE DE FONCTIONS NON DIFFERENTIABLES.

1. Introduction 10
2. Critères d'optimalité 11
3. Approximation optimale de e 12
4. Fonctions d'optimalité 20
5. Dérivées directionnelles 35

CHAPITRE III : DEVELOPPEMENT D'UN ALGORITHME GENERAL THEOREME DE CONVERGENCE.

1. Schéma de l'algorithme général 49
2. Théorème GCAT (théorème de convergence) 50
3. Théorème du pas minimum 52
4. Théorème d'Armijo 55

CHAPITRE IV : APPLICATION A LA RESOLUTION DU PROBLEME INITIAL DE PROGRAMMATION NON LINEAIRE.

1. Objectif 62
2. Principe et méthode de résolution du système des conditions de Karush-Kuhn-Tucker 63
3. Formulation du problème linéaire 75

CHAPITRE V : PROGRAMMATION DE L'ALGORITHME.

1. Organigramme 81
2. Exemples testés 82
3. Commentaires des résultats et conclusion 89

ANNEXE.

BIBLIOGRAPHIE.

CHAPITRE I : DESCRIPTION DU PROBLEME.

§ 1. FORMULATION DU PROBLEME.

Dans notre étude, nous allons nous intéresser à une classe de problèmes non linéaires (PNL) dont la forme générale est la suivante :

PNL
I § 1 (1)

$$\begin{aligned} &\min f(x) \\ &\text{s.c } g(x) \leq 0 \\ &\quad\ h(x) = 0 \end{aligned}$$

où $f : R^n \longrightarrow R$

$g : R^n \longrightarrow R^m$ $\quad f, g, h \in C^2$

$h : R^n \longrightarrow R^l$

Notre but est de trouver un point admissible x du PNL i.e. tel que $g(x) \leq 0$, $h(x) = 0$ qui vérifie les conditions nécessaires d'optimalité de Kuhn-Tucker.

L'optique que nous choisissons est de tenter de résoudre ce problème non linéaire (PNL) par une suite de problèmes linéaires (PL).

à linéariser chaque fonction au voisinage de $\bar{x}$:

$$f(x) \overset{\sim}{=} \tilde{f}(x,\bar{x}) = f(\bar{x}) + (x-\bar{x})' \nabla f(x)$$

$$g_i(x) \overset{\sim}{=} \tilde{g}_i(x,\bar{x}) = g_i(\bar{x}) + (x-\bar{x})' \nabla g_i(\bar{x}), \qquad i = 1,\ldots m$$

$$h_j(x) \overset{\sim}{=} \tilde{h}_j(x,\bar{x}) = h_j(\bar{x}) + (x-\bar{x})' \nabla h_j(\bar{x}), \qquad j = 1,\ldots p$$

Posons $y = x - \bar{x}$ et écrivons chaque composante y_j de y sous la forme $y_j = y_j^+ - y_j^-$ $\qquad j = 1,\ldots n$

où $$y_j^+ = \begin{cases} y_j & \text{si } y_j \geq 0 \\ 0 & \text{sinon} \end{cases}$$

et $$y_j^- = \begin{cases} -y_j & \text{si } y_j \leq 0 \\ 0 & \text{sinon} \end{cases}$$

Comme y_j représente la distance de x_j à $\bar{x}_j$, il est raisonnable de borner cette distance par l'inégalité

$$|y_j| \leq m_j \qquad j = 1,\ldots n$$

où m_j sont des constantes données.

De plus, comme x_j est bornée, on peut combiner les bornes sur y_j et x_j de la façon suivante :

$$0 \leq y_j^+ \leq \min \{m_j, u_j - \bar{x}_j\}$$

$$0 \leq y_j^- \leq \min \{m_j, \bar{x}_j\}$$

Le programme linéaire approximant (Bp) est ainsi donné par

$$(PL_{\bar{x}}) \quad \begin{cases} \min \; f(\bar{x}) + (y^+)' \nabla f(\bar{x}) - (y^-)' \nabla f(\bar{x}) \\ \text{S.c} \; g_i(\bar{x}) + (y^+)' \nabla g_i(\bar{x}) - (y^-)' \nabla g_i(\bar{x}) \geq 0 \quad i = 1,\ldots m \\ h_j(\bar{x}) + (y^+)' \nabla h_j(\bar{x}) - (y^-)' \nabla h_j(\bar{x}) = 0 \quad j = 1,\ldots p \\ 0 \leq y_j^+ \leq \min \{m_j, u_j - \bar{x}_j\} \quad j = 1,\ldots n \\ 0 \leq y_j^- \leq \min \{m_j, \bar{x}_j\} \quad j = 1,\ldots n \end{cases}$$

Procédé algorithmique :

Initialisation : x_o, m_o donnés

résoudre le PL

soit y_1 solution du PL_{x_o}, auquel correspond x_1

$k \leftarrow 1$.

Etape principale : si x_k est admissible pour BP, résoudre le PL_{x_k} en gardant le même m_k

sinon choisir m_{k+1} tq $m_{k+1} < m_k$

résoudre le PL_{x_k} avec m_{k+1} comme bornes

retour en début de étape principale.

L'algorithme se termine si x_k est admissible pour BP et satisfait à certains critères de convergence (p. 463)

Ici le choix des constantes m_j joue un rôle important pour la convergence.

Si les m_j sont trop petits, alors les pas sont petits aussi et donc on progresse plus lentement. Si les m_j sont trop grands, les solutions x_k risquent de ne pas être admissibles pour BP.

Supposons que G C T compact de R^n
où $T = \{x \in R^n \text{ tq } Ax \geqslant b\}$

Schéma algorithmique.

Initialisation : ε donné, suffisamment petit
résoudre le PL

$$\min \; c'x$$
$$\text{s.c. } x \in T$$

soit x^o solution du PL.

$k \leftarrow 0$

aller à l'étape 1.

Etape 1. $G(\varepsilon) = \{x : x \in T \text{ et } g_i(x) \geq -\varepsilon \quad i = 1,1,\ldots 1\}$

Si $x^k \in G(\varepsilon)$ alors STOP

Sinon i.e. $x^k \notin G(\varepsilon)$, alors

$$g_{s_k}(x^k) = \min \{g_i(x^k), \quad i = 0,1,\ldots m\} < 0$$

soit $\nabla g_{s_k}(x^k)$. Aller en 2.

Etape 2. Résoudre le PL.

$$\min \; c'x$$
$$\text{s.c. } g_{s_h}(x^h) + \nabla' g_{s_h}(x^h)\,(x-x^h) \geq 0 \quad h = 0,1,\ldots k$$
$$x \in T$$

soit x^{k+1} solution optimale

poser $k \leftarrow k+1$

aller en 1.

Description de l'algorithme.

Initialisation :

soit $x^o \in X$

$T(x^o) = \{x : g_i(x^o) + (x-x^o)' \nabla g_i(x^o) \geqslant 0 , i = 1,... m\}$

aller à l'étape 1 $k \leftarrow 0$

Etape 1 : considérer le problème convexe suivant :

$$\begin{cases} \min \ c'x \\ \text{s.c. } x \in X(x^k) = T(x^k) \cap U \end{cases}$$

soit x^{k+1} solution de ce problème,

aller à l'étape 2.

Etape 2 : si x^{k+1} satisfait aux conditions de K.T du pb.CCP, alors stop

sinon considérer $T(x^{k+1}) = \{x \in R^n \text{ tq } g_i(x^{k+1}) + (x-x^{k+1})' \nabla g_i(x^{k+1}) \geq 0 , i = 1,... m\}$

$k \leftarrow k+1$

aller à l'étape 1.

La suite engendrée par l'algorithme converge vers un point de K.T de CCP.

Cependant, le choix du point de départ a un effet important sur la localisation du point de K.T pour lequel les solutions du programme convexe convergent.

d) KREUSER a mis en évidence le fait que le système des conditions de K.K.T. du problème initial § 1-1 pouvait faire l'objet d'une minimisation d'une fonction non différentiable telle que la $\| \ \|_1$ ou $\| \ \|_\infty$

Ceci l'a conduit à l'étude de la minimisation d'une classe de fonctions non différentiables.

si et seulement si
$\forall\, x \in D \quad e(\bar{x}) \leq e(x)$

de même, $\bar{x} \in D$ est minimum local de e sur D

si et seulement si
$\exists\, \Omega$ ouvert de R^n tel que $\bar{x} \in \Omega$
et $\forall\, x \in \Omega \cap D \quad e(\bar{x}) \leq e(x)$

Nous dirons aussi que
e(x) satisfait à la condition nécessaire d'optimalité du $n^{\text{ième}}$ ordre en $\bar{x} \in D$
si et seulement si

$$\liminf_{\lambda \overset{>}{\to} 0} \frac{e(\bar{x} + \lambda p) - e(\bar{x})}{\lambda^n} \geq 0 \qquad \forall\, p \in P(\bar{x})$$

où $P(\bar{x}) = \{p \in R^n \text{ tel que } \bar{x} + p \in D;\ p \in \overline{P}\}$

§ 3. APPROXIMATION OPTIMALE DE e.

Comme nous l'avons mentionné au paragraphe 1-3., nous nous proposons d'approximer la fonction e par une fonction convexe appelée " approximation optimale de e ".

3-1. Définition.

Etant donnés la fonction e(x)
le domaine D
l'ensemble P(x) (définis en § 1-1.)

la fonction $\bar{e}(.,.) : R^n \times R^n \longrightarrow R$
$(x,p) \rightsquigarrow \bar{e}(x,p)$
est appelée approximation optimale de e(x) où $x \in D$

$$\left|\frac{e(x + \lambda_1 \bar{p}) - \bar{e}(x,\lambda_1 \bar{p})}{\lambda_1^n}\right| < \tau \frac{|e(x + \lambda_2 \bar{p}) - e(x)|}{\lambda_2^n} + \varepsilon$$

$\forall\ o < \lambda_1 \leqslant \bar{\lambda}$ et $\forall\ o < \lambda_2 \leqslant \bar{\lambda}$

où $n \in \mathbb{N}$.

3-2-2. Définition.

La fonction $\bar{e}(x,p)$ est une approximation optimale du $n^{ième}$ ordre en $e(x)$ où $x \in D$ si et seulement si la définition 3-2-1 est valide $\forall\ p \in P(x)$.

3-2-3. Définition.

La fonction $\bar{e}(x,p)$ est une approximation optimale du $n^{ième}$ ordre de $e(x)$ sur D si et seulement si la définition 3-2-1 est valide $\forall\ x \in D$ et $\forall\ p \in P(x)$.

3-3. Exemples.

Soit $e : R \longrightarrow R$

$x \rightsquigarrow |x|$

Soient $D = R^n$

$P(x) = \bar{P} = \{ p \text{ tel que } |p| \leqslant 1 \}$

$\bar{e}(x,p) = |x + \alpha p|$

Montrons que $\bar{e}\ (.,.)$ est une approximation optimale de $e(.)$

a) $\bar{e}\ (.,.)$ est continue en (x,p) car la fonction $|x|$ est continue.

b) $\bar{e}(.,.)$ est convexe en p.

Il s'agit donc de vérifier que

$$\forall\, x \in D \quad \bar{e}(x,\lambda p + (1-\lambda)q) \leq \lambda \bar{e}(x,p) + (1-\lambda)\bar{e}(x,q)$$
$$\forall\, 0 \leq \lambda \leq 1$$

ou encore par définition de $\bar{e}$,

$$\forall\, x \in D \quad |x + \alpha(\lambda p + (1-\lambda)q)| \leq \lambda |x + \alpha p| + (1-\lambda)|x+\alpha q|$$
$$\forall\, 0 \leq \lambda \leq 1$$

si nous ajoutons et retranchons λx, cela est équivalent à

$$\forall\, x \in D \quad |(\lambda+1-\lambda)x + \alpha(\lambda p) + \alpha(1-\lambda)q| \leq \lambda|x+\alpha p| + (1-\lambda)|x+\alpha q|$$
$$\forall\, 0 \leq \lambda \leq 1$$

ou encore en utilisant l'inégalité triangulaire et comme $0 \leq \lambda \leq 1$

$$\forall\, x \in D \quad |\lambda(x+\alpha p) + (1-\lambda)(x+\alpha q)| \leq \lambda\,|x+\alpha p| + (1-\lambda)|x+\alpha q|$$

c) $\exists\, ?\ 0 < \tau \leq 1$ tel que $\forall\, \varepsilon > 0\ \exists\, \bar{\lambda} > 0$ tel que

$$\frac{||x+\lambda_1 p| - |x+\alpha\lambda_1 p||}{\lambda_1} \leq \tau \frac{||x+\lambda_2 p| - |x||}{\lambda_2} + \varepsilon$$

$$\forall\ 0 < \lambda_1, \lambda_2 \leq \bar{\lambda}$$
$$\forall\, p \in P(x),\ \forall\, x \in D$$

$$\frac{||x+\lambda_1 p| - |x+\alpha\lambda_1 p||}{\lambda_1} \leq \frac{|x+\lambda_1 p - x - \alpha\lambda_1 p|}{\lambda_2}$$

$$\leq \tau \frac{||x+\lambda_2 p| - |x||}{\lambda_2} + \varepsilon \leq \tau \frac{|x+\lambda_2 p - x|}{\lambda_2} + \varepsilon$$

par conséquent, il suffit que τ satisfasse l'inégalité suivante :

$$\frac{|\lambda_1 p(1-\alpha)|}{\lambda_1} \leq \tau \frac{|\lambda_2 p|}{\lambda_2} + \varepsilon$$

en simplifiant, ceci revient à dire que τ satisfait

$$|p||1-\alpha| \leq \tau|p| + \varepsilon \quad \forall\, \varepsilon > 0$$

a) $\overline{e}_\varepsilon(.,.)$ est continue en (x,p)

car $\overline{e}(.,.)$ est continue en (x,p)

$\varepsilon \, \|.\|$ est continue

la somme de 2 fonctions continues est continue.

b) $\overline{e}_\varepsilon(.,.)$ convexe en p

$\forall \, x \in D \quad \overline{e}_\varepsilon(x,\lambda p + (1-\lambda)q) \leq \lambda \, \overline{e}_\varepsilon(x,p) + (1-\lambda) \, \overline{e}_\varepsilon(x,q)$

$\forall \, p,q \in P(x)$

$\forall \, 0 \leq \lambda \leq 1$

ce qui donne, par défintion de $\overline{e}_\varepsilon$

$\forall \, x \in D$

$$\overline{e}(x,\lambda p + (1-\lambda)q) + \varepsilon \, \|\lambda p + (1-\lambda)q\| \leq$$
$$\lambda \, \overline{e}(x,p) + \lambda \, \varepsilon \|p\|$$
$$+ (1-\lambda) \, \overline{e}(x,q) + (1-\lambda) \, \varepsilon \, \|q\|$$
$$\forall \, p,q \in P(x) \text{ et } \forall \, 0 \leq \lambda \leq 1$$

or, $\overline{e}(.,.)$ est approximation optimale et donc convexe en p

de plus, la norme est une fonction convexe.

Ceci implique que :

$\forall \, x \in D$

$$\overline{e}(x,\lambda p + (1-\lambda)q) + \varepsilon \, \|\lambda p + (1-\lambda)q\|$$
$$\leq \underbrace{\lambda \, \overline{e}(x,p)}_{(1)} + \underbrace{(1-\lambda) \, \overline{e}(x,q)}_{(2)} + \underbrace{\varepsilon \, \lambda \, \|p\|}_{(3)} + \underbrace{\varepsilon \, (1-\lambda) \, \|q\|}_{(4)}$$
$$\forall \, q,p \in P(x); \quad \forall \, 0 \leq \lambda \leq 1$$

en rassemblant (1) et (3)

(2) et (4), nous obtenons par définition de $\overline{e}_\varepsilon$

$\forall \, x \in D$

$$\overline{e}_\varepsilon \, (x,\lambda p + (1-\lambda)q) \leq \lambda \, \overline{e}_\varepsilon(x,p) + (1-\lambda) \, \overline{e}_\varepsilon(x,q)$$
$$\forall \, 0 \leq \lambda \leq 1$$
$$\forall \, p,q \in P(x)$$

$$\leq \frac{|\bar{e}(x,\lambda q) - e(x+\lambda q)|}{\lambda} + \varepsilon \|q\|$$

$$\leq \tilde{\tau} \frac{|e(x+\lambda q) - e(x)|}{\lambda} + \varepsilon^* + \varepsilon \frac{\|\lambda q\|}{\lambda} \qquad (I)$$

$$\forall \quad 0 < \lambda \leq \lambda^*$$

$$\text{or} \quad \frac{\varepsilon \|\lambda q\|}{\lambda} \leq (1-\tilde{\tau}-\tau) \frac{|e(x+\lambda q) - e(x)|}{\lambda} \qquad (II) \qquad \text{(par déf. de inf.)}$$

par conséquent, vu la majoration (II), (I) devient :

$$\forall \quad \varepsilon^* > 0 \quad \exists \lambda^* > 0 \quad \text{tq}$$

$$\frac{|\bar{e}_\varepsilon(x,\lambda q) - e(x+\lambda q)|}{\lambda} \leq (1-\tilde{\tau}-\tau+\tilde{\tau}) \frac{|e(x+\lambda p) - e(x)|}{\lambda} + \varepsilon^*$$

$$\forall \quad 0 < \lambda \leq \lambda^*$$

En posant $\tau^* = 1-\tau$, nous obtenons bien la condition c).

3-5. Propriété de l'approximation optimale.

Proposition

Si $e : R^n \longrightarrow R$ est une fonction continue.
$\bar{e}(.,.) : R^n \times R^n \longrightarrow R$ est une approximation optimale de e sur D

alors
$\forall x \in D \quad \bar{e}(x,0) = e(x)$

Preuve :
Comme $\bar{e}(.,.)$ est une approximation optimale de e sur D, la condition c) (cfr 3-1) est vérifiée en $p = 0 \in P(x)$ car $x \in D$. Par conséquent, nous obtenons l'inégalité suivante :

$\exists\ 0 \leq \tau < 1$ tel que $\forall\ \tilde{\varepsilon} > 0\ \exists\ \bar{\lambda} > 0$ tel que

$$\frac{|e(x) - \bar{e}(x,0)|}{\lambda_1} \leq \tau \frac{|e(x) - e(x)|}{\lambda_2} + \tilde{\varepsilon}$$

$\forall\ x \in D$

$\forall\ 0 < \lambda_1, \lambda_2 \leq \bar{\lambda}$

ce qui implique que

$$|e(x) - \bar{e}(x,0)| \leq \tilde{\varepsilon}\ \lambda_1 \qquad \forall\ \tilde{\varepsilon} > 0$$

$\forall\ 0 < \lambda_1 \leq \bar{\lambda}$

et en posant $\tilde{\varepsilon}\ \lambda_1$, il suit que

$$|e(x) - \bar{e}(x,0)| \leq \varepsilon \qquad \forall\ \varepsilon > 0$$

et donc :

$e(x) = \bar{e}(x,0)$

comme x était arbitraire, nous pouvons en déduire que

$\forall\ x \in D : e(x) = \bar{e}(x,0)$

cqfd.

§ 4. FONCTIONS D'OPTIMALITE.

Reprenant le schéma de notre démarche (cfr § 1-3), il nous faut à présent définir ce qu'on entend par fonction d'optimalité.
Son rôle sera d'identifier les points stationnaires et elle constituera un test d'arrêt dans les algorithmes qui seront développés par la suite.

4-1. Définitions.

4-1-1. $\theta : R^n \longrightarrow R$, $x \rightsquigarrow \theta(x)$ est une fonction d'optimalité respectant une condition nécessaire d'optimalité pour $e : R^n \longrightarrow R$ où $x \in D$

si et seulement si elle vérifie les 4 conditions suivantes :

a) $\theta(x)$ est une fonction à valeurs réelles sur D.

b) $\theta(x)$ est semi-continue inférieurement.

c) $\theta(x) \geq 0$

d) si $x^o \in D$, satisfait à la condition nécessaire d'optimalité (cfr § 2) pour $e(x)$ sur D, alors $\theta(x^o) = 0$

4-1-2. Etant donnée une fonction d'optimalité $\theta(x)$, alors le point x^o est appelé point stationnaire ssi $\theta(x^o) = 0$

4-1-3. Multiapplication.

Soient X, Y 2 ensembles; $F : X \longrightarrow \sigma(Y)$

F est une multiapplication sur X dans un sous-ensemble de Y si et seulement si $\forall x \in X$ un sous-ensemble $F(x) \subset Y$ est univoquement défini.

4-1-4. Multiapplication semi-continue supérieurement (SCS).

Etant donnée une multiapplication $F : X \longrightarrow \sigma(Y)$

où $\sigma(Y) = \{ P \subset Y \text{ tel que } \exists x \in X \quad F(x) = P \}$

F est semi-continue supérieurement en $\overline{x} \in X$

ssi $(x^k) \longrightarrow \overline{x}$ où $x^k \in X$

$(y^k) \longrightarrow \overline{y}$ où $y^k \in F(x^k) \subset Y$

alors $\overline{y} \in F(\overline{x})$.

4-1-5. Multiapplication semi-continue inférieurement.

Soit la multiapplication $F : X \longrightarrow \sigma(Y)$

F est semi-continue inférieurement en $\bar{x} \in X$

ssi $\forall\, y \in F(\bar{x})$

$\forall\, (x^i)$ suite de points de X telle que

$(x^i) \xrightarrow[i\to\infty]{} \bar{x}$

on a : $\exists\, k \in N$

$\exists\, (y^i) \longrightarrow y$ telle que $y^i \in F(x^i)$ $i=k, k+1, \ldots$

4-2. Théorème.

Enonçons à présent un théorème qui permet de déterminer quand une fonction est semi-continue inférieurement.

Soit $\eta : U \times V \longrightarrow R$ où $V \subset R^n$

$U \subset R^n$

tq $\eta(u,v)$ est SCI en u et SCI en v.

$\forall\, u \in U$ considérons la multiapplication $V : U \longrightarrow \sigma(V)$

$u \rightsquigarrow V(u)$

avec V(.) semi-continue supérieurement en $\bar{u} \in U$

Supposons que $\forall\, u_{\text{fixé}} \in U$ $\inf_v \{ \eta(u,v) \text{ tq } v \in V(u)\}$

est atteint pour quelques $v \in V(u)$

Soit $V^*(u)$ = ensemble où l'infimum est atteint

$V^*(u) \subset \bar{V} \subset V$ où $\bar{V}$ est un compact.

Alors

$\theta(u) = \inf_v \{\eta(u,v) \text{ tel que } v \in V(u)\}$ est semi-continue inférieurement en u.

Démonstration.

Considérons une suite arbitraire $(u^i)_{i \in \mathbb{N}} \longrightarrow \bar{u} \in U$
où $u^i \in U$.
Associons à chaque $u_i \in U$

$$\inf_v \{\eta(u^i,v) \text{ tq } v \in V(u^i)\}$$

Par hypothèse, nous savons que cet infimum est atteint pour quelques $v^i \in V(u^i)$

$$\eta(u^i,v^i) \overset{not}{=} \inf \{\eta(u^i,v) \text{ tq } v \in V(u^i)\}$$

donc $v^i \in V(u^i)$

Posons $V^*(u^i)$ = l'ensemble des $v \in V(u^i)$ pour lequel l'infimum est atteint.

Par conséquent, $v^i \in V^*(u^i) \subset V(u^i) \subset \bar{V}$

Nous avons ainsi constitué la suite $(v^i)_{i \in \mathbb{N}}$ où $v^i \in V^*(u^i) \ \forall \ i \in \mathbb{N}$

Comme $\bar{V}$ est compact, de la suite $(v^i)_{i \in \mathbb{N}}$, il est possible d'extraire une sous-suite $(\tilde{v}{}^i)_{i \in \mathbb{N}}$ convergeant vers un point $\bar{v} \in \bar{V}$.

Formons la suite correspondante $\{(\tilde{u}{}^i, \tilde{v}{}^i)\}_{i \in \mathbb{N}}$ qui est une sous-suite de $(u^i,v^i)_{i \in \mathbb{N}}$

où $(\tilde{v}{}^i)_{i \in \mathbb{N}} \underset{i \to \infty}{\longrightarrow} \bar{v} \in \bar{V}$

Vu que la suite $(u^i)_{i \in \mathbb{N}} \underset{i \to \infty}{\longrightarrow} \bar{u} \in U$, il en est de même pour la sous-suite $(\tilde{u}{}^i)_{i \in \mathbb{N}}$

$\eta(u,v)$ étant semi-continue inférieurement en u et v, nous

avons que (par définition de SCI § 4-1)

$$\eta(\bar{u},\bar{v}) \leq \liminf_{i \to \infty} \eta(\tilde{u}^i,\tilde{v}^i) \qquad (1)$$

où $(\tilde{u}^i) \longrightarrow \bar{u} \in U$

$(\tilde{v}^i) \longrightarrow \bar{v} \in V$

De plus, V(u) est une multiapplication SCS au point $\bar{u} \in U$

$$\Longrightarrow \bar{v} \in V(\bar{u}) \qquad (2)$$

$$\text{Or} \quad \theta(\bar{u}) = \inf_{v} \{\eta(\bar{u},v) \text{ tq } v \in V(\bar{u})\} \qquad (3)$$

(2) et (3) impliquent $\theta(\bar{u}) \leq \eta(\bar{u},\bar{v}) \qquad (4)$

(1) et (4) impliquent $\theta(\bar{u}) \leq \eta(\bar{u},\bar{v}) \leq \liminf_{i \to \infty} \eta(\tilde{u}^i,\tilde{v}^i)$ (5)

Comme $\bar{v}$ est un point d'accumulation de la suite (v^i) et $v^i \in V$ compact, chaque élément v^i peut être considéré comme faisant partie d'une sous-suite qui converge vers un point d'accumulation de (v^i).

Par conséquent, en vertu de (5), il suit que

$$\theta(\bar{u}) \leq \liminf_{i \to \infty} \eta(u^i,v^i) \qquad (6)$$

Comme $\eta(u^i,v^i) = \inf_{v} \{\eta(u^i,v) \text{ tel que } v \in V(u^i)\}$,

nous obtenons par définition de θ (cfr 4-1-1)

$$\theta(u^i) = \eta(u^i,v^i) \quad \text{où } v^i \in V(u^i) \qquad (7)$$

(6) et (7) entraînent que

$$\theta(\bar{u}) \leq \liminf_{i \to \infty} \theta(u^i) = \liminf_{i \to \infty} \eta(u^i,v^i)$$

cqfd.

Corollaire.

Soit $\eta : U \times V \longrightarrow R$ où $V \subset R^n$
$U \subset R^n$

où $\eta(u,v)$ est SCS en u et v
si $\forall u \in U$ $V(u)$ est une multiapplication avec $V(u) \subset V$
$V(.)$ est SCS en $\bar{u} \in U$

et
si $\forall u_{fixé} \in U$ inf $\{\eta(u,v)$ tel que $v \in V(u)\}$ est atteint pour quelques $v \in V(u)$

Alors
$\theta(u) = -\inf_{v} \{\eta(u,v)$ tel que $v \in V(u)\}$ est semi-continue inférieurement.

Démonstration.

Comme la fonction $-\eta(u,v)$ est SCI, on applique le théorème 4-2 à la fonction $-\eta(u,v)$.

4-3. Théorème.

A partir d'une approximation optimale $\bar{e}$, il nous est possible de définir une fonction d'optimalité.

Soient $e(x)$, D, $\bar{P}$, $P(x)$ comme définis en § 1-1.
$\bar{e}(x,p)$ une approximation optimale (cfr § 3-1).
Alors $\theta(x) = e(x) - \inf_{p \in P(x)} \bar{e}(x,p)$ $\quad x \in D$
est une fonction d'optimalité pour $e(x)$ sur D par rapport à la condition nécessaire d'optimalité du 1er ordre.

Démonstration.

Il suffit de prouver que $\theta(x)$ vérifie les 4 conditions (cfr 4-1) qui définissent une fonction d'optimalité.

a) $\theta(x)$ est une fonction à valeurs réelles sur D car e et $\overline{e}(.,.)$ le sont.

b) $\theta(x)$ est semi-continue inférieurement.
En vue d'appliquer le corollaire du théorème 4-2, montrons que la fonction $\overline{e} : D \times \overline{P} \longrightarrow R$, $(x,p) \rightsquigarrow \overline{e}(x,p)$ satisfait aux hypothèses du corollaire.

1) $\overline{e}$ est semi-continue supérieurement en x et p car $\overline{e}$ est continue en x et p.

2) $\forall x \in D$ $P(x)$ est une multiapplication tq $P(x) \in \overline{P}$.

3) $P(x)$ est semi-continue supérieurement
Soient $(x^k) \longrightarrow \overline{x}$ où $x^k, \overline{x} \in D$

$(p^k) \longrightarrow \overline{p}$ où $p^k \in P(x^k) \subset \overline{P}$

Montrons que $\overline{p} \in P(\overline{x})$:

comme $p^k \in P(x^k) \iff x^k + p^k \in D$

or : $\lim_{k \to \infty} x^k = \overline{x}$, $\lim_{k \to \infty} p^k = \overline{p}$ } par conséquent, $\lim_{k \to \infty} x^k + p^k = \overline{x} + \overline{p}$

de plus, D est fermé $\Longrightarrow \overline{x} + \overline{p} \in D \iff \overline{p} \in P(\overline{x})$

4) $\forall x_{\text{fixé}} \in D$ $\inf_{p \in P(x)} \{ \overline{e}(x,p)$ tel que $p \in P(x)\}$
est atteint en quelques $p \in P(x)$, car $\overline{e}$ est continue sur un compact $P(x) \subset \overline{P}$ et donc atteint ses bornes.

Notons par $P^*(x)$: l'ensemble des points où l'infimum est atteint.

Comme $P^*(x) \subset \overline{P}$, le corollaire du théorème 4-2 nous

permet de conclure que

$\tilde{\theta}(x) = - \inf_{p} \{\bar{e}(x,p)$ tel que $p \in P(x)\}$ est semi-continue inférieurement en x

i.e. par définition de SCI, nous avons

$$\forall x \in D \quad \liminf_{y \to x} \tilde{\theta}(y) \geq \tilde{\theta}(x)$$

en ajoutant e(x) dans les 2 membres de l'inégalité, nous obtenons :

$$\forall x \in D \quad e(x) + \liminf_{y \to x} \tilde{\theta}(y) \geq e(x) + \tilde{\theta}(x)$$

ou encore comme e est continue, nous pouvons écrire

$$\forall x \in D \quad \liminf_{y \to x} e(y) + \liminf_{y \to x} \tilde{\theta}(y) \geq e(x) + \tilde{\theta}(x)$$

Ce qui implique aussi, par majoration :

$$\forall x \in D \quad \liminf_{y \to x} (e(y) + \tilde{\theta}(y)) \geq e(x) + \tilde{\theta}(x)$$

On en déduit que la fonction $e(x) + \tilde{\theta}(x)$ est semi-continue inférieurement.

Or $\theta(x) \equiv e(x) - \inf_{p} \bar{e}(x,p) \equiv e(x) + \tilde{\theta}(x)$. Il résulte donc que $\theta(x)$ est SCI.

c) $\theta(x) \geq 0$

En effet comme $\bar{e}(x, 0) = e(x) \quad \forall x \in D$ (cfr 3.-5)

$0 \in P(x)$ car $x \in D$

par définition de l'infimum, il suit que :

$$- \inf_{p \in P(x)} \bar{e}(x,p) \geq - \bar{e}(x, 0)$$

En ajoutant e(x) dans les 2 membres de l'inégalité, on obtient :

$$\theta(x) = e(x) - \inf_{p \in P(x)} \bar{e}(x,p) \geq e(x) - \bar{e}(x,0) = 0$$

Par conséquent, $\forall x \in D \quad \theta(x) \geq 0$

d) Montrons que si e(x) satisfait à la condition nécessaire d'optimalité du 1er ordre en $\bar{x} \in D$ (cfr 2-1) alors $\theta(\bar{x}) = 0$

Procédons par l'absurde : supposons que $\theta(\bar{x}) > 0$

i.e. $e(\bar{x}) - \inf_{p \in P(\bar{x})} \bar{e}(\bar{x},p) > 0$

par conséquent $\exists\ \bar{p} \in P(\bar{x})$ tq $e(\bar{x}) - \bar{e}(\bar{x},\bar{p}) = \alpha > 0$

puisque $\bar{e}(.,.)$ est une approximation optimale,

$\exists\ 0 \leq \tau < 1$ tq $\forall\ \varepsilon > 0$ $\exists\ \bar{\lambda} > 0$ tq

$$\frac{|\bar{e}(\bar{x},\lambda_1\bar{p}) - e(\bar{x}+\lambda_1\bar{p})|}{\lambda_1} \leq \tau \frac{|e(\bar{x}+\lambda_2\bar{p}) - e(\bar{x})|}{\lambda_2} + \varepsilon$$

$$\forall\ 0 < \lambda_1, \lambda_2 \leq \bar{\lambda}$$

donc en particulier quand $\lambda_1 = \lambda_2 = \lambda$

ce qui nous donne :

$$\frac{|e(\bar{x},\lambda\bar{p}) - e(\bar{x}+\lambda\bar{p})|}{\lambda} \leq \tau \frac{|e(\bar{x}+\lambda\bar{p}) - e(\bar{x})|}{\lambda} + \lambda\varepsilon$$

$$\forall\ 0 < \lambda \leq \bar{\lambda}$$

ou encore

$$\lambda\varepsilon + \tau|e(\bar{x}+\lambda\bar{p}) - e(\bar{x})| \geq |\bar{e}(\bar{x},\lambda\bar{p} + (1-\lambda)0) - e(\bar{x}+\lambda\bar{p})|$$

$$\geq -\bar{e}(\bar{x},\lambda\bar{p} + (1-\lambda)0) + e(\bar{x}+\lambda\bar{p})$$

comme $\bar{e}(x,.)$ est convexe en p, nous obtenons :

$$\lambda\,\varepsilon + \tau\,|e(\bar{x}+\lambda\bar{p}) - e(\bar{x})| \geq -\lambda\bar{e}(\bar{x},\bar{p}) - (1-\lambda)\bar{e}(\bar{x},o) + e(\bar{x}+\lambda\bar{p})$$

$$\geq -\lambda\bar{e}(\bar{x},\bar{p}) - (1-\lambda)\,e(\bar{x}) + e(\bar{x}+\lambda\bar{p})$$

vu que $\bar{e}(\bar{x},0) = e(\bar{x})$

par conséquent, en réarrangeant les termes, l'inégalité devient :

$$e(\bar{x}) - e(\bar{x}+\lambda\bar{p}) + \tau\,|e(\bar{x}+\lambda\bar{p}) - e(\bar{x})|$$

$$\geq \lambda \underbrace{(e(\bar{x}) - \bar{e}(\bar{x},\bar{p}))}_{=\alpha} - \lambda\,\varepsilon$$

$$\forall\ \varepsilon > 0$$

$$\forall\ 0 < \lambda \leq \bar{\lambda}$$

En particulier, l'inégalité est vérifiée quand $\varepsilon = \frac{\alpha}{2}$

$$e(\bar{x}) - e(\bar{x}+\lambda\bar{p}) + \tau\,|e(\bar{x}+\lambda\bar{p}) - e(\bar{x})|$$

$$\geq \lambda\,\alpha - \lambda\,\frac{\alpha}{2} = \lambda\,\frac{\alpha}{2}$$

1er cas : si $e(\bar{x}+\lambda\bar{p}) - e(\bar{x}) > 0$

alors $(\tau-1)\,(e(\bar{x}+\lambda\bar{p}) - e(\bar{x})) \geq \lambda\,\frac{\alpha}{2} > 0$

(impossible)

2e cas : si $e(\bar{x}+\lambda\bar{p}) - e(\bar{x}) \leq 0$

alors $\exists\ \bar{\lambda} > 0$ tq

$(1+\tau)\,(e(\bar{x}) - e(\bar{x}+\lambda\bar{p}) \geq \lambda\,\frac{\alpha}{2}\,\frac{1}{1+\tau} > 0 \quad \forall\ 0 < \lambda \leq \bar{\lambda}$ (1)

Or, par hypothèse e(x) satisfait à la condition nécessaire d'optimalité du 1er ordre en $\bar{x} \in D$.

donc $\liminf_{\lambda \to 0} \dfrac{e(\bar{x}+\lambda p) - e(\bar{x})}{\lambda} \geq 0$ (2)

d'où la contradiction entre (1) et (2).

Par conséquent, comme notre supposition est fausse, il faut donc que $\theta(\bar{x}) = 0$

cqfd.

4-4. Corollaire.

Hypothèses : soient $e(x)$, D, $P(x)$ comme définis en § 1-1.

$\bar{p} \in P(\bar{x})$ tel que $\bar{e}(\bar{x},\bar{p}) < e(\bar{x})$

Thèse : $\exists\, \bar{\lambda} > 0$; $\exists\, \varepsilon > 0$ tq

$e(\bar{x}+\lambda \bar{p}) \leq e(\bar{x}) - \lambda \varepsilon \quad \forall \; 0 \leq \lambda \leq \bar{\lambda}$

Démonstration :

Par hypothèse, nous savons que
$e(\bar{x}) - \bar{e}(\bar{x},\bar{p}) \overset{not}{=} \alpha > 0$ où $\bar{p} \in P(\bar{x})$

Dès lors, il suffit de reprendre les différentes inégalités de la partie d) de la démonstration du théorème 4-3, pour en arriver à la conclusion (1), à savoir : $\exists\, \bar{\lambda} > 0$ tel que

$e(\bar{x}) - e(\bar{x}+\lambda \bar{p}) \geq \lambda \dfrac{\alpha}{2} \dfrac{1}{1+\tau} \quad \forall \; 0 \leq \lambda \leq \bar{\lambda}$

Posons $\varepsilon = \dfrac{\alpha}{2\,(1+\tau)} > 0$

et nous obtenons effectivement que :
$\exists\, \bar{\lambda} > 0 \quad \exists\, \varepsilon > 0$ tq

$e(\bar{x}+\lambda \bar{p}) \leq e(\bar{x}) - \lambda \varepsilon \quad \forall \; 0 < \lambda \leq \bar{\lambda}$

cqfd.

Remarque : ce corollaire exprime le fait que $\bar{p}$ est une direction de descente pour $e(.)$ en $\bar{x}$.

4-5. THéorème.

Ce théorème constitue la réciproque du corollaire 4-4 mais dans le cas plus général d'une approximation optimale du $n^{ième}$ ordre.

Soit $\bar{e}(x,p)$ une approximation optimale de $e(x)$ du $n^{ième}$ ordre en $\bar{x} \in D$

Supposons que $\exists\ \bar{p} \in P(\bar{x})$; $\exists\ \varepsilon > 0$ tq

$$e(\bar{x}+\lambda\bar{p}) \leq e(\bar{x}) - \lambda^n \varepsilon \quad \forall \quad 0 \leq \lambda \leq 1$$

Alors $\exists\ \bar{\lambda} > 0$ et $\exists\ \bar{\varepsilon} > 0$ tq

$$\bar{e}(\bar{x},\lambda\bar{p}) \leq e(\bar{x}) - \lambda^n \bar{\varepsilon} \quad \forall \quad 0 \leq \lambda \leq \bar{\lambda}$$

Démonstration.

Sachant que $\bar{e}(x,.)$ est une approximation optimale de $e(x)$ en $\bar{x} \in D$, $\exists\ 0 \leq \tau < 1$ tq $\forall\ \tilde{\varepsilon} > 0$ $\exists\ \tilde{\lambda} > 0$ tq

$$|e(\bar{x}+\lambda\bar{p}) - \bar{e}(\bar{x},\lambda\bar{p})| \leq \tau\ |e(\bar{x}+\lambda\bar{p}) - e(\bar{x})| + \lambda^n \tilde{\varepsilon}$$

$$\forall \quad 0 < \lambda \leq \tilde{\lambda}$$

et en particulier, quand $\tilde{\varepsilon} = \frac{1-\tau}{2}\varepsilon$

$\exists\ 0 \leq \tau < 1$ tq $\exists\ \tilde{\lambda} > 0$ tq

$$|e(\bar{x}+\lambda\bar{p}) - \bar{e}(\bar{x},\lambda\bar{p})| \leq \tau\ |e(\bar{x}+\lambda\bar{p}) - e(\bar{x})| + \lambda^n \frac{1-\tau}{2}\varepsilon \quad (1)$$

$$\forall \quad 0 < \lambda \leq \tilde{\lambda}$$

Par hypothèse : $e(\bar{x}) - e(\bar{x}+\lambda\bar{p}) \geq \lambda^n \varepsilon > 0 \quad (2)$

$$\forall \quad 0 \leq \lambda \leq 1$$

par conséquent, en enlevant les valeurs absolues et en posant $\bar{\lambda} = \min \{1, \tilde{\lambda}\}$, l'expression (1) devient :

$$|-e(\bar{x}+\lambda\bar{p}) + \bar{e}(\bar{x},\lambda\bar{p})| \leq \tau\,(e(\bar{x}) - e(\bar{x}+\lambda\bar{p}) + \lambda^n\,\frac{(1-\tau)}{2}\,\varepsilon$$

$$\forall \quad 0 < \lambda \leq \bar{\lambda}$$

il suit immédiatement que

$$-e(\bar{x}+\lambda\bar{p}) + e(\bar{x},\lambda\bar{p}) \leq \tau\,(e(\bar{x}) - e(\bar{x}+\lambda\bar{p})) + \lambda^n\,\frac{(1-\tau)}{2}\,\varepsilon$$

$$\forall \quad 0 < \lambda \leq \bar{\lambda}$$

en ajoutant $e(\bar{x})$ dans chaque membre de l'inégalité, nous obtenons

$$e(\bar{x}) - e(\bar{x}+\lambda\bar{p}) + \bar{e}(\bar{x},\lambda\bar{p}) \leq \tau\,(e(\bar{x}) - e(\bar{x}+\lambda\bar{p})) + e(\bar{x}) + \lambda^n\,\frac{(1-\tau)}{2}\,\varepsilon$$

$$\forall \quad 0 < \lambda \leq \bar{\lambda}$$

ce qui nous donne enfin, en réarrangeant les termes

$$(1-\tau)\,\underbrace{(e(\bar{x})-e(\bar{x}+\lambda\bar{p}))}_{\geq\ \lambda^n\,\varepsilon} - \lambda^n\,\frac{(1-\tau)}{2}\,\varepsilon \leq e(\bar{x}) - \bar{e}(\bar{x},\lambda\bar{p})$$

$$\forall \quad 0 < \lambda \leq \bar{\lambda}$$

par la relation (2)

===>

$$(1-\tau)\,\lambda^n\,\varepsilon - \lambda^n\,\frac{(1-\tau)}{2}\,\varepsilon \leq e(\bar{x}) - \bar{e}(\bar{x},\lambda\bar{p})$$

$$\forall \quad 0 < \lambda \leq \bar{\lambda}$$

ou encore

$$\lambda^n\,\frac{(1-\tau)}{2}\,\varepsilon \leq e(\bar{x}) - \bar{e}(\bar{x},\lambda\bar{p}) \qquad \forall \quad 0 < \lambda \leq \bar{\lambda}$$

en choisissant $\bar{\varepsilon} = \frac{1-\tau}{2}\,\varepsilon$, on a la thèse.

cqfd.

4-6. Théorème.

Ce théorème exprime le fait que tout point stationnaire est tel qu'en ce point, la fonction e satisfait à la condition nécessaire d'optimalité.

Soit $\bar{e}(x,p)$ une approximation optimale de e du $n^{ième}$ ordre en $\bar{x} \in D$

Supposons que $\theta(\bar{x}) = e(\bar{x}) - \inf_{p \in P(\bar{x})} \bar{e}(\bar{x},p) = 0$

Alors e(x) satisfait à la condition nécessaire d'optimalité du $n^{ième}$ ordre en $\bar{x} \in P$ (cfr.2-1)

Démonstration.

En raisonnant par l'absurde, supposons que e(x) ne satisfait pas à la condition d'optimalité du $n^{ième}$ ordre en $\bar{x} \in D$.

Par conséquent, $\exists\ \bar{p} \in P(\bar{x})$; $\exists\ \varepsilon > 0$; $\exists\ \bar{\lambda} > 0$ tq

$$\frac{e(\bar{x}+\lambda\bar{p}) - e(\bar{x})}{\lambda^n} \leq -\varepsilon \qquad \forall \quad 0 < \lambda \leq \bar{\lambda}$$

$$\Longrightarrow$$

$$e(\bar{x}+\lambda\bar{p}) - e(\bar{x}) \leq -\lambda^n \varepsilon \qquad \forall \quad 0 < \lambda \leq \bar{\lambda}$$

Le théorème 4-5 nous assure qu'il
$\exists\ \lambda^* > 0$ et $\exists\ \varepsilon^* > 0$ tq
$\bar{e}(\bar{x},\lambda\bar{p}) \leq e(\bar{x}) - \lambda^n \varepsilon^* \qquad \forall \quad 0 < \lambda \leq \lambda^*$

dès lors, nous obtenons que $\bar{e}(\bar{x},\lambda\bar{p}) < e(\bar{x})$
et même $\inf_{p \in P(\bar{x})} \bar{e}(\bar{x},p) \leq e(\bar{x},\lambda\bar{p}) < e(\bar{x})$

ce qui nous amène à dire que

$$\theta(\bar{x}) = e(\bar{x}) - \inf_{p \in P(\bar{x})} \bar{e}(\bar{x},p) > 0$$

d'où la contradiction avec l'hypothèse du théorème.

Conclusion.

Notre supposition étant fausse, nous avons bien que $e(x)$ satisfait à la C.N.O. du $n^{ième}$ ordre en $\bar{x}$.
Ainsi $\bar{x}$ est un "candidat potentiel" au minimum de e sur D.

cqfd.

§ 5. DERIVEES DIRECTIONNELLES.

Dans ce paragraphe, nous généralisons la notion de dérivées aux fonctions non différentiables en introduisant les dérivées directionnelles. Quelques théorèmes établissent le lien entre le signe de la dérivée directionnelle et la possibilité d'obtenir une direction de descente pour e.

5-1. Définitions.

5-1-1.

La fonction réelle e est dite différentiable au pt x dans la direction $p \in R^n$ ($p \neq 0$) s'il existe un réel t et une fonction réelle $\alpha(x,p,.)$ tels que $\forall \lambda \in [0, 1]$

$$e(x+\lambda p) = e(x) + \lambda t + \lambda\alpha(x,p,\lambda)$$

où $\lim_{\lambda \overset{>}{\to} 0} \alpha(x,p,\lambda) = 0$.

Pour rappel, e est différentiable en $\bar{x}$ (au sens classique) s'il existe un voisinage V de $\bar{x}$ tel que $\forall x \in V$ on puisse écrire

$e(x) = e(\bar{x}) + (x-\bar{x})(A + \eta_{x^\circ}(x))$ où $A \in R$ et η_{x° est une fonction telle que $\lim_{x \to x^\circ} \eta_{x^\circ}(x) = 0$

5-1-2.

La dérivée directionnelle de e(.) au pt $\bar{x}$ dans la direction p est définie par

$$\lim_{\lambda \overset{>}{\to} 0} \frac{e(\bar{x}+\lambda p) - e(x)}{\lambda}$$

Si la limite existe, on la note $de(\bar{x},p)$.

De manière analogue, la dérivée (au sens classique) $f'(x^\circ)$ vaut

$$\lim_{h \to 0} \frac{f(x^\circ+h) - f(x^\circ)}{h}$$

5-2. Théorème.

Ce théorème lie les 2 définitions en montrant que la différentiabilité de e implique l'existence de la dérivée directionnelle.

> Si e(.) est différentiable en x dans la direction p $\in R^n$ alors e(.) a une dérivée en x dans la direction p et
>
> $$e(x+\lambda p) = e(x) + \lambda de(x,p) + \lambda\alpha(x,p,\lambda) \qquad \forall \lambda \in]0, 1]$$
>
> où $\lim_{\lambda \to 0^+} \alpha(x,p,\lambda) = 0$

Preuve : Par hypothèse e(.) est différentiable en x dans la direction p i.e.

$\exists$ t $\in$ R et $\exists$ une fonction α tq $\forall \lambda \in]0, 1]$

$$e(x+\lambda p) = e(x) + \lambda t + \lambda\, \alpha(x,p,\lambda)$$

où $\lim_{\lambda \to 0} \alpha(x,p,\lambda) = 0$

Si on soustrait e(x) dans les 2 membres et si l'on divise les 2 membres par λ, on obtient :

$$\frac{e(x+\lambda p) - e(x)}{\lambda} = t + \alpha(x,p,\lambda) \qquad \forall \lambda \in]0, 1]$$

Faisons tendre λ vers 0, il suit que :

$$\lim_{\lambda \to 0} \frac{e(x+\lambda p) - e(x)}{\lambda} = t + \lim \alpha(x,p,\lambda)$$

Mais comme $\lim_{\lambda \to 0} \alpha(x,p,\lambda) = 0$, il résulte que :

$$\lim_{\lambda \to 0} \frac{e(x+\lambda p) - e(x)}{\lambda} = t$$

Puisque par hypothèse t $\in$ R, nous obtenons que :

$$\lim_{\lambda \overset{>}{\to} 0} \frac{e(x+\lambda p) - e(x)}{\lambda} \quad \text{existe et est finie.}$$

notation : $\lim_{\lambda \overset{>}{\to} 0} \frac{e(x+\lambda p) - e(x)}{\lambda} = de\ (x,p)$

cqfd.

Remarque : Si e $\in C^1$ alors

$$de\ (x,p) = \lim_{\lambda \overset{>}{\to} 0} \frac{e(x+\lambda p)-e(x)}{\lambda}$$

$$= \lim_{\lambda \overset{>}{\to} 0} \left(\frac{e(x+\lambda p) - e(x)}{x + \lambda p - x}\right)' \quad p$$

$$= \nabla' e(x)\ p$$

Par conséquent, si e $\in C^1$, on a :

$de(x,p) = \nabla' e(x)\ p$

5-3. Théorème.

Ce théorème montre en quoi le signe de la dérivée directionnelle joue un rôle quant à la possibilité d'avoir une direction de descente pour e(.).

> Si on suppose que e(.) est différentiable en x dans la direction p et si de (x,p) est finie et < 0
> alors $\exists\ \overline{\lambda}$ tel que
> $e(x+\lambda p) < e(x) \qquad \forall\ 0 < \lambda \leq \overline{\lambda}$

Preuve : Puisque par hypothèse e est différentiable en x dans la direction p, on peut utiliser le théorème 5-2.

De là on en déduit :

$$e(x+\lambda p) - e(x) = \lambda de(x,p) + \lambda \alpha(x,p,\lambda) \qquad \forall \lambda \in]0, 1]$$

où $\lim_{\lambda \overset{>}{\to} 0} \alpha(x,p,\lambda) = 0$

Explicitons ce que veut dire $\lim_{\lambda \to 0} \alpha(x,p,\lambda) = 0$

$\lim_{\lambda \overset{>}{\to} 0} \alpha(x,p,\lambda) = 0$ ssi $\forall \varepsilon \in R_o^+ \ \exists \eta \in R^+$ tq

$$|\lambda| < \eta \implies |\alpha(x,p,\lambda)| < \varepsilon$$

Ceci est donc vrai pour $\varepsilon = - de(x,p) > 0$, car $de(x,p) < 0$ par hypothèse.

Par conséquent $\exists \eta \in R^+$ tel que

$$|\lambda| < \eta \implies |\alpha(x,p,\lambda)| < - de(x,p)$$

i.e.

$$|\lambda| < \eta \implies |\alpha(x,p,\lambda)| + de(x,p) < 0$$

ou encore

$$\alpha(x,p,\lambda) + de(x,p) \leq |\alpha(x,p,\lambda)| + de(x,p) < 0$$

Il en résulte que :

$$|\lambda| < \eta \implies \alpha(x,p,\lambda) + de(x,p) < 0$$

On obtient alors :

$$|\lambda| < \eta \implies \lambda(\alpha(x,p,\lambda) + de(x,p))$$

$$= e(x+\lambda p) - e(x) < \lambda \cdot 0 = 0$$

Ainsi en choisissant $\overline{\lambda} = \eta$, il s'ensuit que

$$e(x+\lambda p) - e(x) < 0 \qquad \forall \lambda \in]0, \overline{\lambda}]$$

cqfd.

Interprétation : Si $de(x,p) < 0$, alors p est une direction de descente de e au point x.

Ceci généralise la condition pour que p soit une direction de descente dans le cas où e $\in C^1$
de $(x,p) < 0$ devient $\nabla' e(x)p. < 0$

5-4. Théorème.

Ce théorème constitue une condition nécessaire et suffisante que doit satisfaire tout minimum local de e. Cette condition est à nouveau basée sur le signe de la dérivée directionnelle. Si celui-ci est strictement positif, le théorème nous assure que x est un minimum local moyennant l'hypothèse de différentiabilité.

a) Si e est différentiable en x dans chaque direction $p \in P(x)$ et si e(.) a un minimum local en x sur D, alors
$de(x,p) \geq 0 \qquad \forall\ p \in p(x)$

b) Réciproquement si $de(x,p) > 0 \qquad \forall\ p \in P(x)$ où $p \neq 0$ alors il existe une constante positive c et un voisinage N de x tels que
$e(x+p) - e(x) \geq c\|p\| \qquad \forall\ p \in P(x)$ tq $x+p \in N$

Preuve : Hestenes (1) page 42.

a) Par hypothèse, on a que e(.) est différentiable en x dans la direction p. En utilisant le théorème 5-2, on obtient :

$$e(x+\lambda p) = e(x) + \lambda\, de(x,p) + \lambda\alpha(x,p,\ )$$

$$\forall\ \lambda \in [0,1] \text{ et } \lim_{\substack{\lambda \to 0 \\ >}} \alpha(x,p,\lambda) = 0$$

De plus, par définition d'un minimum local

$\exists$ ouvert Ω de x tel que $e(x) \leq e(\bar{x})$ $\forall$ $\bar{x} \in \Omega \cap D$.

Fixons une direction p arbitrairement. On obtient alors :

$$0 \leq \frac{e(x+\lambda p) - e(x)}{\lambda} = de(x,p) + \alpha(x,p,\lambda)$$

$$\forall\ \lambda \in]0,\bar{\lambda}] \text{ tq } x + \lambda p \in \Omega \cap D,$$

d'où : $0 \leq de(x,p) + \alpha(x,p,\lambda)$ $\forall$ $\lambda \in]0,\bar{\lambda}]$

Si λ tend vers 0, on obtient :

$$0 \leq \lim_{\substack{\lambda \to 0 \\ >}} de(x,p) + \lim_{\substack{\lambda \to 0 \\ >}} \alpha(x,p,\lambda)$$

ou encore vu que $\lim_{\lambda \to 0} \alpha(x,p,\lambda) = 0$

$$0 \leq de(x,p)$$

Comme p a été fixé arbitrairement, il résulte que :

$\forall$ $p \in P(x)$ $\quad de(x,p) \geq 0$

cqfd.

b) Par hypothèse $de(x,p) > 0$ $\forall$ $p \in P(x)$.

Supposons par l'absurde qu'il n'existe pas de constante positive c et de voisinage N de x tels que

$$e(x+p) - e(x) \geq c \,\|p\| \quad \forall \quad p \in P(x) \text{ avec } x + p \in N$$

Cela signifie que $\forall$ entier q $\exists$ $x_q = x + \lambda_q p$ tq

$$e(x_q) - e(x) < \frac{1}{q} \,\|p\| \,\lambda_q$$

Nous pouvons donc former une suite (x_q) convergeant vers x dans la direction p.

On obtient alors que :

$$de(x,p) = \lim_{\substack{\lambda q \to 0 \\ >}} \frac{e(x+p\lambda_q) - e(x)}{\lambda q}$$

$$\leq \lim_{\substack{\lambda \to 0 \\ >}} \frac{\frac{1}{q} \; \| p \| \; \lambda q}{\lambda q} = \lim_{q \to \infty} \frac{1}{q} \; \| p \| = 0$$

Par conséquent, $de(x,p) \leq 0$

ce qui contredit l'hypothèse $de(x,p) > 0$

cqfd

5-5. Théorème.

Celui-ci énonce une condition suffisante sur la fonction e qui permet d'obtenir l'existence des dérivés directionnelles dans toutes les directions.

Si e est convexe et finie en x alors e a une dérivée directionnelle dans chaque direction p $\in R^n$ au point x.

Démonstration.

1ère étape :

D'abord montrons que la fonction $\lambda \longrightarrow \dfrac{e(x+\lambda p) - e(x)}{\lambda}$

est une fonction croissante pour $\lambda > 0$

i.e. $\forall \quad \lambda_1, \lambda_2 \quad tq \quad 0 < \lambda_1 \leq \lambda_2$ on a :

$$\frac{e(x+\lambda_1 p) - e(x)}{\lambda_1} \leq \frac{e(x+\lambda_2 p) - e(x)}{\lambda_2}$$

En effet, comme $0 < \frac{\lambda 1}{\lambda_2} \leq 1$ et étant donné que e est convexe, nous pouvons écrire l'inégalité suivante :

$$e(x+\lambda_1 p) = e\left(\frac{\lambda_1}{\lambda_2}(x+\lambda_2 p) + (1-\frac{\lambda_1}{\lambda_2})x\right)$$

$$\leq \frac{\lambda_1}{\lambda_2} e(x+\lambda_2 p) + (1 - \frac{\lambda_1}{\lambda_2}) e(x)$$

en faisant passer e(x) dans le 1er membre, l'inégalité s'écrit

$$e(x+\lambda_1 p) - e(x) \leq \frac{\lambda_1}{\lambda_2} (e(x+\lambda_2 p) - e(x))$$

Si nous divisons les 2 membres par λ_1, il en résulte :

$$\frac{e(x+\lambda_1 p) - e(x)}{\lambda_1} \leq \frac{e(x+\lambda_2 p) - e(x)}{\lambda_2}$$

Par conséquent, la fonction que nous avons définie est croissante.

2e étape.

$$\lim_{\substack{\lambda \to 0 \\ >}} \frac{e(x+\lambda p) - e(x)}{\lambda} = \lim_{n \to \infty} \frac{e(x+\lambda_n p) - e(x)}{\lambda_n}$$

Par la 1ère étape, $\dfrac{e(x+\lambda_n p) - e(x)}{\lambda_n}$ peut être considéré comme un élément d'une suite décroissante. Par conséquent,

$\lim_{n \to \infty} \dfrac{e(x+\lambda_n p) - e(x)}{\lambda_n}$ existe et est finie car e est finie.

Comme toute suite décroissante tend vers sa borne inférieure, on a :

$$\lim_{n \to \infty} \frac{e(x+\lambda_n p) - e(x)}{\lambda_n} = \inf_{\lambda_n > 0} \frac{e(x+\lambda_n p) - e(x)}{\lambda_n}$$

$$\stackrel{\Delta}{=} de(x,p)$$

En conclusion : $\inf_{\lambda_n > 0} \dfrac{e(x+\lambda_n p) - e(x)}{\lambda_n} = de(x,p)$

Comme p a été choisi de manière arbitraire, nous obtenons que $de(x,p)$ existe et finie $\forall \quad p \in R^n$

cqfd.

Intérêt de ce théorème.

Les fonctions e définies en termes de $\| \ \|_1$ ou $\| \ \|_\infty$, comme il en sera d'ailleurs question par la suite, rentrent dans cette catégorie de fonctions convexes et finies.
De plus, toute approximation optimale $\bar{e}(x,p)$ possède aussi des dérivées directionnelles en p dans chaque direction $q \in R^n$ et celles-ci seront notées $d\bar{e}(x,p,q)$. En effet, toute approximation optimale $\bar{e}(x,p)$ est convexe et finie en p.

Nous pouvons maintenant énoncer un lemme qui établit les relations qui peuvent exister entre les dérivés directionnelles de $e(x)$ et celles de son approximation optimale $\overline{e}(x,p)$.

5-6. Lemme.

Etant donnés $e(x)$ est différentiable en $\overline{x}$ dans la direction $p \in P(\overline{x})$

$\overline{e}(x,p)$ est une approximation optimale de $e(x)$

i) Si $de(\overline{x},p) \neq 0$ et $d\overline{e}(\overline{x},o,p) \neq 0$ alors
$|de(\overline{x},p) - d\overline{e}(\overline{x},o,p)| < |de(\overline{x},p)|$

ii) $de(\overline{x},p) = 0 <===> d\overline{e}(\overline{x},o,p) = 0$

Démonstration de i).

En vertu de la remarque faite en 5-5, nous pouvons affirmer que $\overline{e}(x,p)$ a une dérivée directionnelle en 0 dans la direction p, i.e. $d\overline{e}(\overline{x},o,p)$ existe et est finie

$$|de(\overline{x},p) - d\overline{e}(\overline{x},o,p)| = \text{cfr déf. 5-1-2}$$

$$= \left| \lim_{\lambda \to 0^{>}} \frac{e(\overline{x}+\lambda p) - e(\overline{x})}{\lambda} - \lim_{\lambda \to 0^{>}} \frac{\overline{e}(\overline{x},\lambda p) - \overline{e}(\overline{x},o)}{\lambda} \right|$$

Comme $\bar{e}(.,.)$ est une approximation optimale, $\bar{e}(\bar{x},o) = e(\bar{x})$ (cfr.II. 3-5), nous obtenons :

$$= \left| \lim_{\substack{\lambda \to 0 \\ >}} \frac{(e(\bar{x}+\lambda p) - \bar{e}(\bar{x},\lambda p))}{\lambda} \right|$$

$$= \lim_{\substack{\lambda \to 0 \\ >}} \frac{\left| e(\bar{x}+\lambda p) - \bar{e}(\bar{x},\lambda p) \right|}{\lambda}$$

$$\leq \quad \tau \lim_{\substack{\lambda \to 0 \\ >}} \frac{\left| e(\bar{x}+\lambda p) - e(\bar{x}) \right|}{\lambda} + \varepsilon$$

par déf. de l'appr.opt

$$\forall \quad \varepsilon > 0$$

cfr II.3-1
$0 \leq \tau < 1$

$$= \quad \tau \left| \lim_{\substack{\lambda \to 0 \\ >}} \frac{e(\bar{x} +\lambda p) - e(\bar{x})}{\lambda} \right| + \varepsilon$$

$$\forall \quad \varepsilon > 0$$

$$= \quad \tau \, |de(\bar{x},p)| + \varepsilon \qquad \forall \quad \varepsilon > 0$$

En faisant tendre $\varepsilon \to 0$, on obtient que

(1) $\quad |de(\bar{x},p) - d\bar{e}(\bar{x},o,p)| \leq \tau \, |de(\bar{x},p)|$

et comme $de(\bar{x},p) \neq 0$ et $d\bar{e}(\bar{x},o,p) \neq 0$, il suit que

$|de(\bar{x},p) - d\bar{e}(\bar{x},o,p)| < |de(\bar{x},p)|$ vu que $\tau < 1$.

Démonstration de ii) ===>

Si $de(\bar{x},p) = 0$ alors l'inégalité (1) devient

$$|\underbrace{de(\bar{x},p)}_{=0} - d\bar{e}(\bar{x},o,p)| \leq 0$$

ce qui permet de conclure immédiatement que $d\bar{e}(\bar{x},o,p) = 0$

<===

Si $d\bar{e}(\bar{x},o,p) = 0$, alors l'inégalité (1) s'écrit

$$|de(\bar{x},p)| \leq \tau\ |de(\bar{x},p)|$$

par suite,

$$(1-\tau)\ |de(\bar{x},p)| \leq 0$$

Par conséquent, $de(\bar{x},p) = 0$

cqfd.

5-7. Théorème d'équivalence.

Soit $\bar{e}(.,.)$ une approximation optimale de $e(.)$
Si $e(.)$ est différentiable en x dans la direction
$p \in P(x)$, alors les énoncés suivants sont équivalents :

a) $d\bar{e}(x,o,p) < 0$

b) $de(x,p) < 0$

c) $\exists\ \varepsilon > 0\ \exists\ \bar{\lambda} > 0$ tq $\forall\ 0 < \lambda \leq \bar{\lambda}$

$\bar{e}(x,\lambda p) - e(x) \leq -\lambda\ \varepsilon$

d) $\exists\ \varepsilon > 0\ \exists\ \bar{\lambda} > 0$ tq $\forall\ 0 < \lambda \leq \bar{\lambda}$

$e(x+\lambda p) - e(x) \leq -\lambda\ \varepsilon$

Démonstration :

a) ===> b)

Supposons par l'absurde que $de(x,p) \geq 0$,et distinguons 2 cas.

1er cas : $d\bar{e}(x,o,p) < 0$ et $de(x,p) = 0$

dès lors, par le lemme 5-6, nous avons que $d\bar{e}(x,o,p) = 0$
Ce qui contredit l'hypothèse " $d\bar{e}(x,o,p) < 0$ ".

2ème cas : $d\bar{e}(x,o,p) < 0$ et $de(x,p) > 0$

On en déduit que

$$|de(x,p) - d\bar{e}(x,o,p)| = de(x,p) - d\bar{e}(x,o,p)$$
$$> de(x,p)$$
$$= |de(x,p)|$$

Nous obtenons donc que

$$|de(x,p) - d\bar{e}(x,o,p)| > |de(x,p)|$$

Ceci contredit le lemme 5-6.

Conclusion : on ne pouvait supposer $de(x,p) \geq 0$ et par conséquent $de(x,p) < 0$

b) ===> a)

Supposons à nouveau par l'absurde que $d\bar{e}(x,o,p) \geq 0$.
Par conséquent

$$|de(x,p) - d\bar{e}(x,o,p)| = -de(x,p) + d\bar{e}(x,o,p)$$
$$\geq - de(x,p)$$
$$= |de(x,p)|$$

Nous obtenons aussi que

$$|de(x,p) - d\bar{e}(x,o,p)| \geq |de(x,p)|$$

ce qui contredit le lemme 5-6.
Donc il ne fallait pas supposer que $d\bar{e}(x,o,p) < 0$

b) ===> d)

Par hypothèse e est différentiable en x dans la direction $p \in P(x)$ et $de(x,p) < 0$. En appliquant le théorème II-§5-3,

$\exists\, \bar{\lambda} > 0$ tq $e(x+\lambda p) - e(x) < 0 \quad \forall\ 0 < \lambda \leq \bar{\lambda}$

ou encore

$\exists\ \bar{\lambda} > 0$ tq $\exists\ \varepsilon' > 0$ tq $e(x+\lambda p) - e(x) \leq -\varepsilon'$ $\forall\ 0 < \lambda \leq \bar{\lambda}$

ou encore

$\exists\ \bar{\lambda} > 0$ et $\exists\ \varepsilon = \varepsilon'/\bar{\lambda}$ tq $e(x+\lambda p) - e(x) \leq -\bar{\lambda}\ \varepsilon$ $\forall\ 0 < \lambda \leq \bar{\lambda}$

et comme $\lambda \leq \bar{\lambda}$,ceci implique que

$e(x+\lambda p) - e(x) \leq -\bar{\lambda}\ \varepsilon \leq -\lambda\ \varepsilon$

et donc $e(x+\lambda p) - e(x) \leq -\lambda\ \varepsilon$ $\forall\ 0 < \lambda \leq \bar{\lambda}$

d) ===> b)

Nous avons que $\exists\ \varepsilon > 0\ \exists\ \bar{\lambda} > 0$ tq $\forall\ 0 < \lambda \leq \bar{\lambda}$

$e(x+\lambda p) - e(x) \leq -\lambda\ \varepsilon < 0$

En divisant par λ, l'inégalité s'écrit

$$\frac{e(x+\lambda p) - e(x)}{\lambda} \leq -\varepsilon < 0 \qquad \forall\ 0 < \lambda \leq \bar{\lambda}$$

dès lors,

$$\lim_{\lambda \overset{>}{\to} 0} \frac{e(x+\lambda p) - e(x)}{\lambda} \leq -\varepsilon < 0$$

Ce qui signifie que $de(x,p) \leq -\varepsilon < 0$

De manière similaire, on démontre que

a) <===> c) en remplaçant dans b) <===> d) $e(x+\lambda p)$ par $\bar{e}(x,\lambda p)$

et par transitivité on obtient les 4 équivalences.

cqfd.

CHAPITRE III. DEVELOPPEMENT D'UN ALGORITHME GENERAL. THEOREME DE CONVERGENCE.

Ce chapitre présente un algorithme général pour résoudre le problème de minimisation posé au chapitre II, § 1, à savoir

$$\text{Minimiser } e(x) \quad x \in D$$

Ensuite, nous étudierons la convergence de cet algorithme et déterminerons une règle pour la recherche linéaire.

§ 1. SCHEMA DE L'ALGORITHME GENERAL.

Soient $e(x)$, D, $P(x)$ ainsi qu'ils ont été définis en II. § 1.

Soit $\bar{e}(x,p)$ une approximation optimale (cfr II. § 3-1).

Soit $\theta(x)$ une fonction d'optimalité de e (cfr II. § 4-1).

Initialisation.

Etant donné $x_o \in D$
poser $k = o$
aller à l'étape 1.

Etape 1.

déterminer $p^k \in P(x^k)$ tel que
$e(x^k) - \bar{e}(x^k,p^k) \geq \sigma\ (\theta(x^k))$

où $\sigma : [o,\infty[\longrightarrow [o,\infty[$ est une fonction continue,

strictement croissante et telle que $\sigma(0) = 0$
aller à l'étape 2.

Etape 2.

Si $\theta(x^k) = 0$ STOP : arrêt sur un point stationnaire
sinon faire une recherche linéaire qui consiste à trouver
$x^{k+1} = x^k + \lambda^k p^k$ selon une règle qui répond aux conditions suivantes : $x^{k+1} \in D$

si $(x^k j, p^k j) \xrightarrow{j \to \infty} (\bar{x}, \bar{p})$

alors $e(\bar{x}) - \bar{e}(\bar{x}, \bar{p}) = 0$

retour à l'étape 1.

§ 2. THEOREME GCAT (Théorème de convergence de l'algorithme).

La suite $(x^k)_k$ engendrée par l'algorithme se termine sur un point stationnaire i.e $\theta(x^{\bar{k}}) = 0$
ou
si la suite $(x^k)_k$ a un point d'accumulation $\bar{x}$
alors $\bar{x}$ est point stationnaire i.e $\theta(\bar{x}) = 0$

Démonstration.

1er cas $\exists\ \bar{k}$ tel que $e(x^{\bar{k}}) - \bar{e}(x^{\bar{k}}, p^{\bar{k}}) = 0$

dès lors, par l'étape 1 de l'algorithme, nous avons que
$0 = e(x^{\bar{k}}) - \bar{e}(x^{\bar{k}}, p^{\bar{k}}) \geq \sigma(\theta(x^{\bar{k}})) \geq 0$

par conséquent, $\sigma(\theta(x^{\bar{k}})) = 0$

et comme σ est strictement croissante, on en déduit que
$\theta(x^{\bar{k}}) = 0$
ce qui signifie que $x^{\bar{k}}$ est un point stationnaire.

2ème cas. Supposons $\theta(x^k) \neq 0 \quad \forall\, k$

Vu la stricte croissance de σ et comme $\sigma(0) = 0$, ceci implique que $e(x^k) - \bar{e}(x^k, p^k) \geq \sigma(\theta(x^k)) > 0 \quad \forall\, k$

Par hypothèse, la suite $(x^k)_k$ admet un point d'accumulation $\bar{x}$.

De plus $\forall\, k \quad p^k \in p(x^k) \subset \bar{P}$

$\forall\, k \quad p^k \in \bar{P}$ qui est un compact.

Par conséquent, la suite $(p^k)_k$ admet un point d'accumulation $\bar{p} \in \bar{P}$

Par définition d'un point d'accumulation, la suite $(p^k)_k$ possède une sous-suite $(p^{kj})_j$ convergeant vers $\bar{p}$

Choisissons de même une sous-suite $(x^{kj})_j$ telle que $(x^{kj}) \xrightarrow[j \to \infty]{} \bar{x}$

Conclusion : $(x^{kj}) \longrightarrow \bar{x}$

$(p^{kj}) \longrightarrow \bar{p}$

et en vertu de l'étape 2 de l'algorithme, nous obtenons que $e(\bar{x}) - \bar{e}(\bar{x}, \bar{p}) = 0$

ce qui entraîne que :

$0 = e(\bar{x}) - \bar{e}(\bar{x}, \bar{p}) \geq \sigma(\theta(\bar{x})) \geq 0$

et donc $\sigma(\theta(\bar{x})) = 0$

comme σ est strictement croissante, nous en déduisons que $\theta(\bar{x}) = 0$

i.e $\bar{x}$ est point stationnaire.

§ 3. THEOREME DU PAS MINIMUM.

Soient $\overline{e}(x,p)$ une approximation optimale de $e(x)$ (cfr II.3-1).
la suite $(x^k)_k$ engendrée par l'algorithme III. § 1.

Considérons $\overline{\lambda} \in]0,1]$

Si λ^k satisfait à la relation suivante :

$$e(x^k) - e(x^k+\lambda^k p^k) = e(x^k) - \inf_{0 < \lambda \leq \overline{\lambda}} e(x^k+\lambda p^k)$$

et

si (x^{k_j},p^{k_j}) converge vers $(\overline{x},\overline{p})$

alors $e(\overline{x}) - \overline{e}(\overline{x},\overline{p}) = 0$

Démonstration.

Par définition de l'infimum, nous aurons que :

$\forall \lambda \in [0,\overline{\lambda}]$:

$$e(x^{k_j} + \lambda p^{k_j}) \geq \inf_{0 \leq \lambda' \leq \overline{\lambda}} e(x^{k_j+\lambda'} p^{k_j}) = e(x^{k_j+\lambda^{k_j}} p^{k_j}) = e(x^{k_j+1})$$

$$\Longrightarrow e(x^{k_j+\lambda p^{k_j}}) \geq e(x^{k_j+1}) \quad \forall \lambda \in [0,\overline{\lambda}]$$

Comme la suite engendrée par l'algorithme satisfait l'étape 1, on en déduit que $e(x^k) \geq e(x^{k+1})$, ce qui traduit que la suite $(e(x^k))$ est décroissante et comme

$k_j < k_{j+1}$

$k_j + 1 \leq k_{j+1}$, nous pouvons écrire que

$$e(x^{k_j}) \geq e(x^{k_j+1}) \geq e(x^{k_j+2}) \geq \geq e(x^{k_j+1})$$

Vu que $e(x)$ est continue et que la suite $(x^{k_j+\lambda p^{k_j}}) \longrightarrow \overline{x}+\lambda\overline{p}$,

$$\lim_{j\to\infty} e(x^{k_j+\lambda p^{k_j}}) \geq \lim_{j\to\infty} (e(x^{k_j+1})) \quad \forall \ 0 \leq \lambda \leq \overline{\lambda}$$

ou encore

$$(1) \quad e(\bar{x}+\lambda\bar{p}) \geq e(\bar{x}) \qquad \forall \ \lambda \in]0,\bar{\lambda}[$$

$$\text{i.e.} \quad e(\bar{x}+\lambda\bar{p}) - e(\bar{x}) \geq 0 \qquad \forall \ \lambda \in]0,\bar{\lambda}[$$

de plus, par le choix de p^k dans l'algorithme, nous avons que

$$\left.\begin{array}{l} p^k \in P(x^k) \\ \text{tq} \quad p^{k_j} \longrightarrow \bar{p} \\ x^{k_j} \longrightarrow \bar{x} \end{array}\right\} (2)$$

Etant donné que la multiapplication qui envoie x^{k_j} sur $P(x^{k_j})$ est SCS (cfr dém. du théorème II.4-3), les conditions (2) nous permettent d'affirmer que

$$\bar{p} \in P(\bar{x}) \quad (3) \quad \text{(cfr II.4-1-4)}$$

Reprenant les relations (1) et (3) et en passant à la limite dans (1), nous obtenons

$$\lim_{\substack{\lambda \to 0 \\ >}} \frac{e(\bar{x}+\lambda\bar{p}) - e(\bar{x})}{\lambda} \overset{\Delta}{=} de\ (\bar{x},\bar{p}) \geq 0$$

en appliquant le théorème d'équivalences (cfr II. § 4-7), nous en déduisons

$$de\ (\bar{x},\bar{p}) \geq 0 \quad <==> \quad d\bar{e}(\bar{x},o,\bar{p}) \geq 0$$

et par définition de $d\bar{e}(\bar{x},o,\bar{p})$, cela signifie que

$$\lim_{\substack{\lambda \to 0 \\ >}} \frac{\bar{e}(\bar{x},\lambda\bar{p}) - e(\bar{x})}{\lambda} \geq 0$$

En conséquence,

$$(4)\ \exists\ \tilde{\lambda} > 0 \quad \text{tq} \quad \frac{\bar{e}(\bar{x},\lambda\bar{p}) - e(\bar{x})}{\lambda} \geq 0 \quad \forall\ \lambda \in\ ]0\ \tilde{\lambda}\ ]$$

Distinguons maintenant 2 cas suivant la valeur de $\tilde{\lambda}$

1er cas : $\tilde{\lambda} \geq 1$

Choisissons dans (4) : $\lambda = 1$; l'inégalité (4) s'écrit

$$\bar{e}(\bar{x},\bar{p}) - e(\bar{x}) \geq 0 \qquad (5)$$

2e cas : $0 < \tilde{\lambda} < 1$.

dès lors, $\forall\ \lambda \in\ ]0, \tilde{\lambda}]$ et par (4) et par convexité de $\bar{e}(.,.)$

$$0 \leq \frac{\bar{e}(\bar{x},\lambda\ \bar{p}) - e(\bar{x})}{\lambda} \leq \frac{\lambda\bar{e}(\bar{x},\bar{p}) + (1-\lambda\ )e(\bar{x}) - e(\bar{x})}{\lambda}$$

$$= \frac{\lambda\ (\ \bar{e}(\bar{x},\bar{p}) - e(\bar{x})\ )}{\lambda}$$

$$= \bar{e}(\bar{x},\bar{p}) - e(\bar{x})$$

et donc $\bar{e}(\bar{x},\bar{p}) - e(\bar{x}) \geq 0 \qquad (6)$

Par l'étape 1) de l'algorithme III§1, nous savons que

$$e(x^{kj}) - \bar{e}(x^{kj},p^{kj}) \geq 0$$

En passant à la limite, l'inégalité reste encore vraie i.e.

$$\lim_{j \to \infty}\ (\ e(x^{kj}) - \bar{e}(x^{kj},p^{kj})\ ) \geq 0$$

Il suit que $e(\bar{x}) - \bar{e}(\bar{x},\bar{p}) \geq 0 \qquad (7)$

Regroupant les inégalités (5), (6) et (7), nous obtenons que

$$\left.\begin{array}{l} \bar{e}(\bar{x},\bar{p}) - e(\bar{x}) \geq 0 \\ e(\bar{x}) - \bar{e}(\bar{x},\bar{p}) \geq 0 \end{array}\right\} \Longrightarrow e(\bar{x}) - \bar{e}(\bar{x},\bar{p}) = 0$$

cqfd.

§ 4. THEOREME D'ARMIJO.

Celui-ci établit la convergence de l'algorithme III. § 1 lorsqu'on utilise la règle d'Armijo

Soit $\bar{e}(x,p)$ une approximation optimale de $e(x)$

Soit $\alpha > 0$; soit la suite $(x^k)_k$ engendrée par l'algorithme III. § 1.

Considérons λ^k le plus grand élément parmi $\{1, \frac{1}{2}, \frac{1}{4}, ..\}$ tel que $e(x^k) - e(x^k+\lambda^k p^k) \geq \alpha(\lambda^k)^2 \ (e(x^k) - \bar{e}(x^k,p^k))$

Alors si $(x^{k_j}, p^{k_j}) \longrightarrow (\bar{x},\bar{p})$

on a $e(\bar{x}) - \bar{e}(\bar{x},\bar{p}) = 0$

Démonstration.

PAS 1.

Montrons que $\exists\ \tau \in]0,1]$ tq $\forall\ \varepsilon > 0\ \ \exists\ \bar{\lambda} > 0$ tq

$$e(x^k) + e(x^k+\lambda p^k) \geqslant \frac{\lambda}{2}\ (e(x^k, p^k)) - \frac{\lambda}{2}\ \varepsilon$$

$$\forall\ 0 < \lambda \leq \bar{\lambda}$$

On suppose $\theta(x^k) > 0$ car sinon $\theta(x^k) = 0$ et on a la thèse.

Comme $\bar{e}(x,p)$ est une approximation optimale de e en x^k,

$\exists\ \tau \in]0,1]\ \ \forall\ \varepsilon > 0\ \ \exists\ \tilde{\lambda} > 0$ tq

$$\frac{|e(x^k+\lambda p^k) - e(x^k,\lambda p^k)|}{\lambda} \leq \tau\ \frac{|e(x^k+\lambda p^k) - e(x^k)|}{\lambda} + \varepsilon$$

$$\forall\ \lambda \in\ ]0,\tilde{\lambda}]$$

De plus, par le choix de p^k dans l'algorithme,

$\bar{e}(x^k,p^k) - e(x^k) \leq -\sigma(\theta(x^k)) < 0$

car $\theta(x^k) > 0$ et la fonction σ est strictement croissante.

En appliquant le théorème II.4-5, on obtient

$\exists\ \hat{\lambda} > 0$ tel que

$e(x^k+\lambda p^k) < e(x^k) \quad \forall \quad 0 < \lambda \leq \hat{\lambda}$

Posons $\bar{\lambda} = \min\{\tilde{\lambda},\hat{\lambda}, 1\}$

On en déduit, par convexité de $\bar{e}$, que $\forall \quad 0 < \lambda \leq \bar{\lambda}$

$$0 < \lambda(e(x^k) - \bar{e}(x^k,p^k)) \leq e(x^k) - \bar{e}(x^k,p^k)$$
$$= e(x^k) - \bar{e}(x^k,p^k) + e(x^k+\lambda p^k) - e(x^k+\lambda p^k)$$
$$\leq (1+\tau)\ (e(x^k) - e(x^k+\lambda p^k)) + \lambda\ \varepsilon$$

ou encore

$$\lambda(e(x^k) - \bar{e}(x^k,p^k)) \leq (1+\tau)(e(x^k) - e(x^k+\lambda p^k)) + \lambda\ \varepsilon$$
$$\forall \quad 0 < \lambda \leq \bar{\lambda}$$

ce qui peut encore s'écrire comme suit :

$$(1+\tau)\ (e(x^k) - e(x^k+\lambda p^k)) \geq \lambda(e(x^k) - \bar{e}(x^k,p^k)) - \lambda\ \varepsilon$$
$$\forall \quad 0 < \lambda \leq \bar{\lambda}$$

Puisque $\tau \in]0,1]$, $(1+\tau) \leq 2$. Par conséquent :

$$(1) \quad e(x^k) - e(x^k+\lambda p^k) \geq \frac{\lambda}{2}\ (e(x^k) - \bar{e}(x^k,p^k)) - \frac{\lambda}{2}\ \varepsilon$$
$$\forall \quad 0 < \lambda \leq \bar{\lambda}$$

PAS 2.

Montrons qu'il est possible de déterminer λ^k le plus grand élément $\in \{1, \frac{1}{2}, \frac{1}{4}, \ldots\}$ qui satisfasse à la règle d'Armijo i.e.

$$e(x^k) - e(x^k+\lambda^k p^k) \geq \alpha(\lambda^k)^2 \ (e(x^k) - \bar{e}(x^k,p^k))$$

L'inégalité (1) est valable quel que soit $\varepsilon > 0$. Prenons

$$\varepsilon \leq 2\alpha \ (e(x^k) - \bar{e}(x^k,p^k))$$

Soit λ^k tel que $\lambda^k \in \{1, \frac{1}{2}, \frac{1}{4}, \ldots\}$ et tel que

$$\varepsilon \leq (1-\lambda^k)\ 2\alpha \ (e(x^k) - \bar{e}(x^k,p^k))$$

et $$\varepsilon > (1-2\lambda^k)\ 2\alpha \ (e(x^k) - \bar{e}(x^k,p^k))$$

Deux cas peuvent se présenter.

1er cas : $\lambda^k \leq \bar{\lambda}$

posons $\varepsilon^k = \varepsilon$

2e cas : $\lambda^k > \bar{\lambda}$

Par conséquent, il faut augmenter ε. Prenons $\bar{\lambda}^k$ le plus grand élément $\in \{1, \frac{1}{2}, \frac{1}{4}, \ldots\}$ tel que $\bar{\lambda}^k \leq \bar{\lambda}$

Posons $\varepsilon^k = (1-\bar{\lambda}^k)\ 2\alpha \ (e(x^k) - \bar{e}(x^k,p^k))$

ce qui implique :

$$\varepsilon^k > (1-2\bar{\lambda}^k)\ 2\alpha \ (e(x^k) - \bar{e}(x^k,p^k)$$

Ainsi on est ramené au 1er cas.

Majorons $-\varepsilon^k$ par $-(1-\lambda^k)\ 2\alpha \ (e(x^k) - \bar{e}(x^k,p^k))$ que nous remplaçons dans l'inégalité (1), ce qui nous donne

$$e(x^k) - e(x^k+\lambda^k p^k) \geq \frac{\lambda}{2}^k \ (1-1+\lambda^k)\ 2\alpha \ (e(x^k) - \bar{e}(x^k,p^k))$$

i.e.

$$e(x^k) - e(x^k+\lambda^k p^k) \geq \alpha \ (\lambda^k)^2 \ (e(x^k) - \bar{e}(x^k,p^k))$$

en posant $x^{k+1} = x^k + \lambda^k p^k$, il en résulte

$$e(x^k) - e(x^{k+1}) \geq \alpha(\lambda^k)^2 \, (e(x^k) - \bar{e}(x^k, p^k)) > 0 \qquad (2)$$

PAS 3. Montrons que

pour toute sous-suite $(x^{k_j})_j$ telle que $(x^{k_j}) \longrightarrow \bar{x}$

et $(p^{k_j}) \longrightarrow \bar{p}$, on a

$$\lim_{j \to \infty} (\lambda^{k_j})^2 \, (e(x^{k_j}) - \bar{e}(x^{k_j}, p^{k_j})) = 0$$

Par (2) on a $e(x^k) > e(x^{k+1})$. Ce qui signifie que la suite $(e(x^k))_k$ est une suite décroissante.

Il en est de même pour toute sous-suite de $(e(x^k))_k$.

La suite $(e(x^{k_j}))_j$ est aussi une suite décroissante

où $k_j \leq k_{j+1}$. Il suit que

$$e(x^{k_j}) > e(x^{k_j+1}) > \ldots > e(x^{k_j+1})$$

Nous obtenons que $e(x^{k_j}) - e(x^{k_j+1}) > 0$

Par hypothèse, la suite $(x^{k_j})_j$ converge vers $\bar{x}$ et $e(.)$ est finie dans D et continue, il s'ensuit que :

$$(e(x^{k_j}) - e(x^{k_j+1})) \underset{j \to \infty}{\longrightarrow} 0$$

Dès lors, en passant aussi à la limite dans l'inégalité (2) pour la suite $(x^{k_j})_j$

$$\begin{aligned} 0 &= \lim_{j \to \infty} (e(x^{k_j}) - e(x^{k_j+1})) \\ &\geq \lim_{j \to \infty} (e(x^{k_j}) - e(x^{k_j+1})) \\ &\geq \lim_{j \to \infty} \alpha(\lambda^{k_j})^2 \, (e(x^{k_j}) - \bar{e}(x^{k_j}, p^{k_j})) \geq 0 \end{aligned}$$

D'où $\lim_{j \to \infty} \alpha(\lambda^{k_j})^2 \, (e(x^{k_j}) - \overline{e}(x^{k_j}, p^{k_j})) = 0$

En divisant par α, nous obtenons

(3) $\lim_{j \to \infty} (\lambda^{k_j})^2 \, (e(x^{k_j}) - \overline{e}(x^{k_j}, p^{k_j})) = 0$.

PAS 4.

Montrer que pour toute sous-suite $(x^{k_j})_j$ telle que $(x^{k_j}) \longrightarrow \overline{x}$ et $(p^{k_j}) \longrightarrow \overline{p}$, il en résulte que $e(\overline{x}) - \overline{e}(\overline{x}, \overline{p}) = 0$

1er cas. Supposons $\lambda^{k_j} \geq \delta > 0 \quad \forall \; k_j$

Dans (3), nous avons

$$0 = \lim (\lambda^{k_j})^2 \, (e(x^{k_j}) - \overline{e}(x^{k_j}, p^{k_j}))$$

$$\geq \lim_{j \to \infty} \delta^2 (e(x^{k_j}) - \overline{e}(x^{k_j}, p^{k_j}))$$

Vu que $e(x^{k_j}) - \overline{e}(x^{k_j}, p^{k_j}) \geq 0$, nous obtenons finalement

(4) $\lim_{j \to \infty} (e(x^{k_j}) - \overline{e}(x^{k_j}, p^{k_j})) = 0$

Or, les suite $(x^{k_j})_j$ et (p^{k_j}) convergent respectivement vers $\overline{x}$ et $\overline{p}$ et comme les fonctions e et $\overline{e}$ sont continues, nous pouvons affirmer que $(e(x^{k_j})) \longrightarrow e(\overline{x})$

$$\overline{e}(x^{k_j}, p^{k_j}) \longrightarrow \overline{e}(\overline{x}, \overline{p})$$

Dès lors, $\lim_{j \to \infty} e(x^{k_j}) - \overline{e}(x^{k_j}) - \overline{e}(x^{k_j}, p^{k_j}) = e(\overline{x}) - \overline{e}(\overline{x}, \overline{p})$

et par (4) , il en résulte que $e(\overline{x}) - \overline{e}(\overline{x}, \overline{p}) = 0$

2e cas : Supposons $(\lambda^{k_j}) \xrightarrow[j\to\infty]{} 0$ et qu'il existe $\delta > 0$ tel que

(5) $\alpha(e(x^{k_j}) - \bar{e}(x^{k_j},p^{k_j})) \geq \delta > 0 \quad \forall \quad k_j$

(6) Comme $(\lambda^{k_j}) \longrightarrow 0$, on peut supposer que $\lambda^{k_j} < \frac{1}{4}$ à partir d'un certain rang.

De plus, ε^{k_j} a été choisi de telle sorte que

$$\varepsilon^{k_j} > 2\alpha(1-2\lambda^{k_j})\ (e(x^{k_j}) - \bar{e}(x^{k_j},p^{k_j}))$$

les relations (5), (6), (7) impliquent :

$$\varepsilon^{k_j} > 2(1-2\lambda^{k_j})\ \delta > \delta \qquad \forall \quad k_j \text{ tel que } \lambda^{k_j} < \frac{1}{4}$$

Ainsi nous obtenons $\varepsilon^{k_j} > \delta \qquad \forall \quad k_j$ tq $\lambda^{k_j} < \frac{1}{4}$

Comme $\bar{e}(.,.)$ est une approximation optimale,

$0 \leq \tau < 1$ tq $\exists\ \bar{\lambda} > 0$ tel $\forall \quad 0 < \lambda \leq \bar{\lambda}$

$$\frac{|e(x^{k_j}+\lambda p^{k_j}) - \bar{e}(x^{k_j},\lambda p^{k_j})|}{\lambda} \leq \tau \frac{|e(x^{k_j}+\lambda p^{k_j}) - e(x^{k_j})|}{\lambda} + \delta$$

$$\leq \tau \frac{|e(x^{k_j}+\lambda p^{k_j}) - e(x^{k_j})|}{\lambda} + \varepsilon^{k_j}$$

$(\varepsilon^{k_j} > \delta)$

ou encore

$$\frac{|e(x^{k_j}+\lambda p^{k_j}) - \bar{e}(x^{k_j},p^{k_j})|}{\lambda} \leq \tau \frac{|e(x^{k_j}+\lambda p^{k_j}) - e(x^{k_j})|}{\lambda} + \varepsilon^{k_j} \qquad (8)$$

$\forall \quad 0 < \lambda \leq \bar{\lambda}$

De même $\lambda^{kj} > 0$ tq la condition c) (cfr.II.3-2) est vérifiée pour $\varepsilon^{kj} > 0$.

Donc $\bar{\lambda}^{kj} \geq \bar{\lambda}$ car $\varepsilon^{kj} > \delta$.

L'inégalité (8) est donc vérifiée au moins pour tous les $\lambda \leq \bar{\lambda}$ et éventuellement pour des $\lambda > \bar{\lambda}$.

Soit $\tilde{\lambda}$ le plus grand élément $\in \{1, \frac{1}{2}, \frac{1}{4}, \ldots\}$ tq $\tilde{\lambda} \leq \bar{\lambda}$

(9) et tq $\lambda^{kj} \geq \tilde{\lambda}$ $\quad \forall \; k_j$

On a raisonné sur un ε^{kj} arbitraire. Prenons ε^{kj} tq $\lambda^{kj} < \tilde{\lambda}$

λ^{kj} existe car $\tilde{\lambda}$ fixé et $\lambda^{kj} \xrightarrow[j \to \infty]{} 0$

Par conséquent, $\exists\, \varepsilon^{kj}$ tq $\lambda^{kj} < \tilde{\lambda}$ (10)

mais par (9), on a $\lambda^{kj} \geq \tilde{\lambda}$

D'où on a établi une contradiction.

Il s'ensuit que si $(\lambda^{kj}) \longrightarrow 0$, il n'existe pas de $\delta > 0$ tq $\alpha(e(x^{kj}) - \bar{e}(x^{kj}, p^{kj})) \geq \delta$

Ce qui permet de conclure que si (λ^{kj}) converge vers 0,

$$\lim_{j \to \infty} \alpha\, e(x^{kj}) - \bar{e}(x^{kj}, p^{kj})) = 0$$

Selon un raisonnement analogue dans le ler cas, on en déduit que si $(x^{kj}) \longrightarrow \bar{x}$ et $(p^{kj}) \longrightarrow \bar{p}$ alors $e(\bar{x}) - \bar{e}(\bar{x}, \bar{p}) = 0$

cqfd.

CHAPITRE IV : APPLICATION A LA RESOLUTION DU PROBLEME INITIAL DE PROGRAMMATION NON LINEAIRE.

§ 1. OBJECTIF.

Dans ce chapitre, nous nous proposons de particulariser l'algorithme présenté en III.-§1 à la résolution du problème posé au chapitre I.-§1.-(1). Pour rappel, ce problème consiste à trouver un point de Kuhn-Tucker du PNL suivant :

IV.§1.(1)

$$\begin{aligned} &\text{Min } f(x) \\ &\text{s.c. } g(x) \leq 0 \\ &\qquad h(x) = 0 \end{aligned}$$

où $f : R^n \longrightarrow R^n$
$g : R^n \longrightarrow R^m$
$h : R^n \longrightarrow R^1$

telles que $f,g,h \in C^2$ et satisfont à la condition de de Lipschitz jusqu'à l'ordre 2

i.e. $\| \nabla_2 f(x+p) - \nabla_2 f(x) \| \leq M_1 \|p\|$

et $\| \nabla^2 f(x+p) - \nabla^2 f(x) \| \leq M_2 \|p\|$

où M_1, M_2 sont des constantes positives

et $p \in \{ p \in R^n$ tel que $\|p\| \leq 1 \}$

et x appartient à un sous-ensemble compact de R^n.

De même pour les fonctions g et h.

En d'autres termes, il s'agit de trouver un point $z = (x,u,v) \in R^{n+m+1}$ qui satisfait aux conditions de Karush-Kuhn-Tucker du problème (1) qui sont

$$\nabla f(x) + \sum_{i=1}^{m} \nabla g_i(x) u_i + \sum_{j=1}^{l} \nabla h_j(x) v_j = 0$$

$$g_i(x) \leq 0 \qquad i=1,\ldots m$$

$$h_j(x) = 0 \qquad j=1,\ldots l$$

$$u_i \geq 0 \qquad i=1,\ldots m$$

$$u'g(x) \geq 0$$

IV.§1.(2)

notation : $\nabla L(z) = \nabla f(x) + \sum_{i=1}^{m} \nabla g_i(x) u_i + \sum_{j=1}^{l} \nabla h_j(x) v_j$

§ 2. PRINCIPE ET METHODE DE RESOLUTION DU SYSTEME DES CONDITIONS DE KARUSH-KUHN-TUCKER.

Le système IV.§1.(2) peut encore être transformé en un problème général de complémentarité linéaire :

$$\nabla L(z) = 0$$

$$g_i(x) \leq 0 \qquad i=1,\ldots m$$

$$h_j(x) = 0 \qquad j=1,\ldots l$$

$$u_i \geq 0 \qquad i=1,\ldots m$$

$$u'g(x) + v'h(x) - x'\nabla L(z) \geq 0$$

IV.§2.(3)

Nous constatons immédiatement que les 2 systèmes sont équivalents dans le sens où si l'un a une solution, l'autre admet aussi une solution qui est la même.

La résolution du système (3) va ici faire l'objet d'une minimisation. Regroupons les informations de ce système (3) en définissant la fonction e(z) de la manière suivante :

$$e(z) = \| \nabla L(z),\ (g(x))_+,\ h(x),\ (-u)_+,\ (-u'g(x) - v'h(x) + x'\nabla L(z))_+ \|_\infty$$

ou (autre possibilité)

$$e(z) = \| \nabla L(z),\ (g(x))_+, h(x), (-u)_+,\ (-u'g(x)-v'h(x)+x'\nabla L(z))_+ \|_1$$

où $(g_i(x))_+ = \max \{ 0,\ g_i(x)\}$ et $z=(x,u,v) \in R^{n+m+1}$

Choisissons notre domaine $D = R^{n+m+1}$
Avec ce choix de e(z), nous avons que e(z) est une fonction continue sur D mais non différentiable vu qu'il s'agit d'une norme. Si nous considérons la minimisation de e(z) sur D i.e.

$$\min_{z \in R^{n+m+1}} e(z)$$

nous pouvons énoncer la proposition suivante

S'il existe un $z^* \in R^{n+m+1}$ tq le minimun soit atteint i.e. $e(z^*) = 0$ alors z^* est un point de K.K.T.

Preuve. pour le cas de $\| \ \|_\infty$
Si $e(z^*) = 0$, alors chaque composante de la $\| \ \|_\infty$ est nulle.

Ce qui nous donne

IV.§2.(4)
$$\left[\begin{array}{l} \nabla L(z^{\ast}) = 0 \\ (g(x^{\ast}))_{+} = 0 \\ h(x^{\ast}) = 0 \\ (-u^{\ast})_{+} = 0 \\ (-u^{\ast\prime} g(x^{\ast}) - v^{\ast\prime} h(x^{\ast}) + x^{\ast\prime} \nabla L(z))_{+} = 0 \end{array}\right.$$

où $z^{\ast} = (x^{\ast}, u^{\ast}, v^{\ast})$

$(g(x^{\ast}))_{+} = 0 \quad <\Rightarrow \max\{0, g(x^{\ast})\} = 0 <=> g(x^{\ast}) \leq 0$

$(-u^{\ast})_{+} = 0 \quad <=> \max\{0, -u^{\ast}\} = 0 \quad <=> \quad u^{\ast} \geq 0$

$(-u^{\ast\prime} g(x^{\ast}) - v^{\ast\prime} h(x^{\ast}) + x^{\ast\prime} \nabla L(z))_{+} = 0$

$==> u^{\ast\prime} g(x^{\ast}) + v^{\ast\prime} h(x^{\ast}) - x^{\ast\prime} \nabla L(z) \geq 0$

Par conséquent, nous obtenons

IV.§2.(5)
$$\boxed{\begin{array}{l} \nabla L(z^{\ast}) = 0 \\ g(x^{\ast}) \leq 0 \\ h(x^{\ast}) = 0 \\ u^{\ast\prime} g(x^{\ast}) + v^{\ast\prime} h(x^{\ast}) - x^{\ast\prime} \nabla L(z) \geq 0 \\ u^{\ast} \geq 0 \end{array}}$$

De ce fait, $z^{\ast}$ est solution du système (3). Ce qui signifie bien qu'il est un point de Kuhn-Tucker du problème (1) (cfr IV.§1).

Comme dans les chapitres II et III, à la fonction e(z) nous allons associer une approximation optimale $\bar{e}(z,s)$

Pour cela, écrivons le système IV.§2.(3) au point (z+s)

où $z = (x,u,v) \in R^{n+m+1}$

$s = (p,q,r) \in R^{n+m+1}$

Ainsi le système s'écrit

IV.§2.(6)

$$\nabla L(z+s) = 0$$

$$g(x+p) \leq 0$$

$$h(x+p) = 0$$

$$(u+q) \geq 0$$

$$(u+q)'g(x+p) + (v+r)'h(x+p) - (x+p)'\nabla L(z+s) \geq 0$$

Lorsqu'on linéarise le système IV.§2.(6) au pt z+s :
(en négligeant les termes du second ordre en p,q,r)

$$\text{a)} \quad \nabla L(z+s) = \nabla f(x+p) + \sum_{i=1}^{m} \nabla g_i(x+p)(u_i+q_i) + \sum_{j=1}^{l} \nabla h_j(x+p)(v_j+r_j)$$

$$\simeq \nabla f(x) + \nabla^2 f(x)p + \sum_{i=1}^{m} (\nabla g_i(x) + \nabla^2 g_i(x)p)(u_i+q_i) + \sum_{j=1}^{l} (\nabla h_j(x) + \nabla^2 h_j(x)p)(v_j+r_j)$$

$$\simeq \nabla f(x) + \sum_{i=1}^{m} \nabla g_i(x)u_i + \sum_{j=1}^{l} \nabla h_j(x)v_j + (\nabla^2 f(x) + \sum_{i=1}^{m} \nabla^2 g_i(x)u_i + \sum_{j=1}^{l} \nabla^2 h_j(x)v_j)p + \nabla' g(x)q + \nabla' h(x)r$$

$$= \nabla L(z) + \nabla^2 L(z)p + \nabla' g(x)q + \nabla' h(x)r$$

b) $g(x+p) = g(x) + \nabla g(x)p$

c) $h(x+p) = h(x) + \nabla h(x)p$

d) $u + q$

e) $(u+q)'g(x+p) + (v+r)'h(x+p) - (x+p)'\nabla L(z+s)$

$\tilde{=} (u+q)'(g(x) + \nabla g(x)p) + (v+r)'(h(x) + \nabla h(x)p)$

$-(x+p)'(\nabla L(z) + \nabla^2 L(z)p + \nabla' g(x)q + \nabla' h(x)r)$

$\tilde{=} u'g(x) + u'\nabla g(x)p + q'g(x) + q'\nabla g(x)p + v'h(x)$
$+ v'\nabla h(x)p + r'h(x) + r'\nabla h(x)p - x'\nabla L(z) - x'\nabla^2 L(z)p$
$-x'\nabla' g(x)q - x'\nabla' h(x)r - p'\nabla L(z) - p'\nabla^2 L(z)p - p'\nabla' g(x)q$
$-p'\nabla' h(x)r$

$\tilde{=} u'g(x) + u'\nabla g(x)p + g'(x)q + h'(x)v + v'\nabla h(x)p + h'(x)r$
$- x'\nabla L(z) - x'\nabla^2 L(z)p - x'\nabla' g(x)q - x'\nabla' h(x)r - \nabla' L(z)p$

Ainsi, la linéarisation du système IV.§2.(6) nous conduit au sytème suivant :

$$
\begin{aligned}
&\nabla L(z) + \nabla^2 L(z)p + \nabla' g(x)q + \nabla' h(x)r = 0 \\
&g(x) + \nabla g(x)p \leq 0 \\
&h(x) + \nabla h(x)p = 0 \\
&u + q \geq 0 \\
&u'g(x) + u'\nabla g(x)p + g'(x)q + h'(x)v + v'\nabla h(x)p \\
&+h'(x)r - x'\nabla L(z) - x'\nabla^2 L(z)p - x'\nabla' g(x)q \\
&- x'\nabla' h(x)r - \nabla' L(z)p \geq 0
\end{aligned}
\qquad \text{IV.§2.(7)}
$$

A partir de ce système, nous définissons la fonction $\bar{e}(z,s)$ comme suit:

IV. §2.(8)

$$\begin{aligned}\overline{e}(z,s) = \| & \nabla L(z) + \nabla^2 L(z)p + \nabla' g(x)q + \nabla' h(x)r, \\ & (g(x) + \nabla g(x)p)_+, \; h(x) + \nabla h(x)p, \\ & (-u-q)_+, \\ & (-u'g(x) - v'h(x) + x'\nabla L(z) - u'\nabla g(x)p \\ & - v'\nabla h(x)p + \nabla' L(z)p + x'\nabla^2 L(z)p - q'g(x) \\ & + q'\nabla g(x)x - r'h(x) + r'\nabla h(x)x)_+ \|_\infty\end{aligned}$$

ou (autre possibilité suivant le choix de e(z))

$\overline{e}(z,s) =$ || mêmes composantes $||_1$

A partir de maintenant, nous utiliserons la $\| . \|_\infty$, sachant que les raisonnements qui vont être faits peuvent se traduire de manière analogue dans le cas de la $\| \; \|_1$.

Montrons que la fonction $\overline{e}(z,s)$ que nous venons de définir est une approximation optimale de e(z) sur R^{n+m+1}.
Il suffit de vérifier les 3 conditions établies en II.§3.1.

$D = R^{n+m+1}$

$\overline{P} = P(z) = \{ \; s \in R^{n+m+1} \quad tq \quad \| s \|_\infty \leq 1 \; \}^{not} P$

a) $\overline{e}(.,.)$ est continue en (z,s)
car la fonction norme est continue
à valeurs réelles
$< + \infty$

b) Montrons que $\overline{e}(z,s)$ est convexe en s. Ceci suivra par convexité de la $\| \; \|_\infty$

i.e.

$$\overline{e}(z,\lambda s+(1-\lambda)t) \overset{?}{\leq} \lambda \, \overline{e}(z,s) + (1-\lambda)\overline{e}(z,t)$$

$\forall \quad 0 \leq \lambda \leq 1$
$\forall \; s,t \in P$

Preuve :

soient $s = (p_1, q_1, r_1)$

$t = (p_2, q_2, r_2)$

$$\bar{e}(z, \lambda s + (1-\lambda)t) =$$

$$\| \lambda \nabla L(z) + (1-\lambda)\nabla L(z) + \nabla^2 L(z)(\lambda p_1 + (1-\lambda)p_2) + \nabla' g(x)(\lambda q_1 + (1-\lambda)q_2)$$
$$+ \nabla' h(x)(\lambda r_1 + (1-\lambda)r_2),$$
$$(\lambda g(x) + (1-\lambda)g(x) + \nabla g(x)(\lambda p_1 + (1-\lambda)p_2))_+,$$
$$\lambda h(x) + (1-\lambda)h(x) + \nabla h(x)(\lambda p_1 + (1-\lambda)p_2),$$
$$(-\lambda u - (1-\lambda)u - \lambda q_1 - (1-\lambda)q_2)_+,$$
$$(\lambda(-u'g(x) - v'h(x) + x'\nabla L(z)) + (1-\lambda)(-u'g(x) - v'h(x) + x'\nabla L(z))$$
$$-u'\nabla g(x)(\lambda p_1 + (1-\lambda)p_2) - v'\nabla h(x)(\lambda p_1 + (1-\lambda)p_2)$$
$$+ \nabla' L(z)(\lambda p_1 + (1-\lambda)p_2) + x'\nabla^2 L(z)(\lambda p_1 + (1-\lambda)p_2)$$
$$-(\lambda q_1 + (1-\lambda)q_2)'g(x) + (\lambda q_1 + (1-\lambda)q_2)'\nabla g(x)x$$
$$-(\lambda r_1 + (1-\lambda)r_2)'h(x) + (\lambda r_1 + (1-\lambda)r_2)'\nabla h(x)x)_+ \|_\infty$$

En procédant composante par composante et en utilisant l'inégalité triangulaire pour la $|\ |$, nous obtenons

$$\bar{e}(z, \lambda s + (1-\lambda)t) \leq$$
$$\| \lambda \nabla L(z) + \nabla^2 L(z)\lambda p_1 + \lambda \nabla' g(x) q_1 + \lambda \nabla' h(x) r_1,$$
$$\lambda g(x) + \lambda \nabla g(x) p_1, \lambda h(x) + \nabla h(x)\lambda p_1, (-\lambda u - \lambda q_1)_+,$$
$$(\lambda(-u'g(x) - v'h(x) + x'\nabla L(z)) - u'\nabla g(x)\lambda p_1 - v'\nabla h(x)\lambda p_1$$
$$+ \nabla' L(z)\lambda p_1 + x'\nabla^2 L(z)\lambda p_1 - \lambda q_1' g(x) + \lambda q_1'\nabla g(x)x$$
$$-\lambda r_1' h(x) + \lambda r_1'\nabla h(x)x)_+ \|_\infty$$
$$+ \| (1-\lambda)\nabla L(z) + \nabla^2 L(z)(1-\lambda)p_2 + (1-\lambda)\nabla' g(x) q_2 + (1-\lambda)\nabla' h(x) r_2,$$
$$(1-\lambda)g(x) + (1-\lambda)\nabla g(x) p_2 + (1-\lambda)h(x) + \nabla h(x)(1-\lambda)p_2,$$
$$(-(1-\lambda)u - (1-\lambda)q_2)_+, ((1-\lambda)(-u'g(x) - v'h(x) + x'\nabla L(z)),$$
$$-(1-\lambda)u'\nabla g(x)p_2 - (1-\lambda)v'\nabla h(x)p_2 + (1-\lambda)\nabla' L(z)p_2 +$$
$$(1-\lambda)x'\nabla^2 L(z)p_2 - (1-\lambda)q_2' g(x) + (1-\lambda)q_2'\nabla g(x)x$$
$$-(1-\lambda)r_2' h(x) + (1-\lambda)r_2'\nabla h(x)x)_+ \|_\infty$$

En mettant λ en évidence dans le 1er terme du second membre de l'inégalité et $(1-\lambda)$ dans le 2ème terme, on obtient que le 2ème membre de l'inégalité vaut exactement

$\lambda\, \bar{e}(z,s) + (1-\lambda)\bar{e}(z,t)$

On en conclut que la condition b) est satisfaite.

c) $\exists\ ?\ 0 \leq \tau < 1$ tel que $\forall\ \varepsilon > 0$ $\exists\ \bar{\lambda} > 0$ tel que

$$\frac{|e(z+\lambda s) - \bar{e}(z,\lambda s)|}{\lambda} \leq \tau \frac{|e(z+\lambda s) - e(z)|}{\lambda} + \varepsilon$$

$\forall\ 0 < \lambda \leq \bar{\lambda}$

$\forall\ z \in R^{n+m+1}$

$\forall\ s \in P$

Preuve : par définition de e et de $\bar{e}$, nous avons que

$$\frac{|e(z+\lambda s) - \bar{e}(z,\lambda s)|}{\lambda} =$$

$$\Big| \| \nabla L(z+\lambda s),\ (g(x+\lambda p))_+, h(x+\lambda p),\ (-u-\lambda q)_+,\ ((-u-\lambda q)'g(x+\lambda p) + (-v-\lambda r)'h(x+\lambda p) + (x+\lambda p)'\nabla L(z+\lambda s))_+ \|_\infty$$

$$- \| \nabla L(z) + \nabla^2 L(z)\lambda p + \nabla' g(x)\lambda q + \nabla' h(x)\lambda r,\ (g(x) + \nabla g(x)\lambda p)_+,$$

$$h(x) + \nabla h(x)\lambda p,\ (-u-\lambda q)_+,\ (-u'g(x) - v'h(x) + x'\nabla L(z)$$

$$-u'\nabla g(x)\lambda p - v'\nabla h(x)\lambda p + \nabla' L(z)\lambda p + x'\nabla^2 L(z)\lambda p - \lambda q'g(x)$$

$$+ \lambda q'\nabla g(x)x - \lambda r'h(x) + \lambda r'\nabla h(x)x)_+ \|_\infty \Big| \frac{1}{\lambda}$$

Sachant que $| \|a\| - \|b\| | \leq \|a - b\|$, nous obtenons la majoration suivante vu que $(a_+ - b_+) \leq (a - b)_+$:

$$\leq \| \nabla L(z+\lambda s) - \nabla L(z) - \nabla^2 L(z)\lambda p - \nabla' g(x)\lambda q - \nabla' h(x)\lambda r,$$

$$(g(x+\lambda p) - g(x) - \nabla g(x)\lambda p)_+,\ h(x+\lambda p) - h(x) - \nabla h(x)\lambda p,$$

$$(-u - \lambda q + u + \lambda q)_+,$$

$$((-u-\lambda q)'g(x+\lambda p) + (-v-\lambda r)'h(x+\lambda p) + (x+\lambda p)'\nabla L(z+\lambda s)$$
$$+ u'g(x) + v'h(x) - x'\nabla L(z) + u'\nabla g(x)\lambda p + v'\nabla h(x)\lambda p$$
$$- \nabla'L(z)\lambda p - x'\nabla^2 L(z)\lambda p + \lambda q'g(x) - \lambda q'\nabla g(x)x + \lambda\, r'h(x)$$
$$- \lambda r'\nabla h(x)x)_+ \Big|\Big|_\infty \frac{1}{\lambda}$$

Procédons composante par composante :

$$1) \quad \frac{|| h(x+\lambda p) - h(x) - \nabla h(x)\lambda p ||}{\lambda} = \quad (\star\, 1)$$

Comme $h \in C^2$, en appiquant la formule de la moyenne :

$\exists$ cte $c(\lambda)$ $0 < c(\lambda) < \lambda$ tq $h(x+\lambda p) - h(x) = \nabla h(x+c(\lambda)p)\lambda\, p$

Par conséquent, $(\star\, 1)$ peut s'écrire comme suit :

$$= \frac{|| (\nabla h(x+c(\lambda)p) - \nabla h(x))\lambda p ||_\infty}{\lambda} \leq M_1 \; || c(\lambda)p || \;\; || p ||$$

h est localement Lipschitz.

En passant à la limite lorsque $\lambda \longrightarrow 0$, nous obtenons que $c(\lambda) \longrightarrow 0$.

et donc

$$\lim_{\substack{\lambda \to 0 \\ >}} \frac{|| h(x+\lambda p) - h(x) - \nabla h(x)\lambda p ||}{\lambda} = 0$$

De plus h est continue en x et donc

$\exists$ $\bar{\lambda} > 0$ tq $h(x+\lambda p)$ et $h(x)$ sont du même signe $\forall\ 0 \leq \lambda \leq \bar{\lambda}$

Par conséquent, $\big|\, || h(x+\lambda p) || - || h(x) ||\, \big| = || h(x+\lambda p) - h(x) ||$

$\forall \quad 0 \leq \lambda \leq \bar{\lambda}$

Dès lors $\exists\ 0 \leq \tau_1 < 1$ tq $\forall\ \varepsilon > 0$ $\exists\ \bar{\lambda}_1$ suffisamment petit tq

$$\frac{|| h(x+\lambda p) - h(x) - \nabla h(x)\lambda p ||}{\lambda} \leq \tau_1 \frac{|| h(x+\lambda p) - h(x) ||}{\lambda} + \varepsilon$$

$$\forall \quad 0 < \lambda \leq \bar{\lambda}_1$$

2) $\| g(x+\lambda p) - g(x) - \nabla g(x)\lambda p \| = (\divideontimes 2)$

Il suffit de suivre le même raisonnement car $g \in C^2$ et est local. Lipschitz.

Par conséquent $\exists\ 0 \leq \tau_2 < 1$ tq $\forall\ \varepsilon > 0\ \exists\ \bar{\lambda}_2 > 0$ tq

$$\frac{\| g(x+\lambda p) - g(x) - \nabla g(x)\lambda p \|}{\lambda} \leq \tau_2 \frac{\| g(x+\lambda p) - g(x) \|}{\lambda} + \varepsilon$$

$\forall\ 0 < \lambda \leq \bar{\lambda}_2$

3) $$\frac{\| \nabla L(z+\lambda s) - \nabla L(z) - \nabla^2 L(z)\lambda p - \nabla' g(x)\lambda q - \nabla' h(x)\lambda r \|}{\lambda} = (\divideontimes 3)$$

Comme $f,g,h \in C^2$, on en déduit que $\nabla L(z) \in C^1$ et en appliquant à nouveau la formule de la moyenne, on trouve

$\exists\ 0 < c(\lambda) < \lambda$ tq

$$\nabla L(z+\lambda s) - \nabla L(z) - \nabla^2 L(z)\lambda p - \nabla' g(x)\lambda q - \nabla' h(x)\lambda r =$$

$$(\nabla^2 L(z+c(\lambda)s)\lambda p + \nabla' g(x+c(\lambda)p)\lambda q + \nabla' h(x+c(\lambda)p)\lambda r$$

$$- \nabla^2 L(z)\lambda p - \nabla' g(x)\lambda q - \nabla' h(x)\lambda r)$$

Comme f,g,h sont local. Lipschitz jusqu'à l'ordre 2, il suit que

$$(\divideontimes 3) \leq M_2 \| c_2(\lambda)p \| \|q\| + M_1 \| c_1(\lambda)p \| \|r\| + M_3 \| c_3(\lambda)p \| \|p\|$$

De manière analogue, $\exists\ 0 \leq \tau_3 < 1$ tq $\forall\ \varepsilon > 0\ \exists\ \bar{\lambda}_3 > 0$ tq

$$\frac{\| \nabla L(z+\lambda s) - \nabla L(z) - \nabla^2 L(z)\lambda p - \nabla' g(x)\lambda q - \nabla' h(x)\lambda r \|}{\lambda}$$

$$\leq \tau_3 \frac{\| \nabla L(z+\lambda s) - \nabla L(z) \|}{\lambda} + \varepsilon$$

$\forall\ 0 < \lambda \leq \bar{\lambda}_3$

4) $\| (-u-\lambda q)'g(x+\lambda p) + (-v-\lambda r)'h(x+\lambda p) + (x+\lambda p)'\nabla L(z+\lambda s)$

$+ u'g(x) + v'h(x) - x'\nabla L(z) + u'\nabla g(x)\lambda p + v'\nabla h(x)\lambda p$

$- \nabla'L(z)\lambda p - x'\nabla^2 L(z)\lambda p + \lambda q'g(x) - \lambda q'\nabla g(x)x + \lambda r'h(x)$

$- \lambda r'\nabla h(x)x \|_\infty \frac{1}{\lambda} = \quad (\ *4\ )$

En réarrangeant les termes, on obtient que

$$(\ *4\ ) = \| -u'(g(x+\lambda p) - g(x) - \nabla g(x)\lambda p) - v'(h(x+\lambda p) - h(x) - \nabla h(x)\lambda p) - \lambda q'(g(x+\lambda p) - g(x)) - \lambda r'(h(x+\lambda p) - h(x)) + \lambda q'(\nabla L(z+\lambda s) - \nabla L(z)) + x'(\nabla L(z+\lambda s) - \nabla L(z) - \nabla^2 L(z)\lambda p - \nabla' g(x)\lambda q - \nabla' h(x)\lambda r) \|_\infty \frac{1}{\lambda}$$

En utilisant les majorations sur les termes obtenues en 1), 2), 3), il suit que

$$(\ *4\ ) \leq \| u \| M_2 c_2(\lambda) \| p \|^2 + \| v \| M_1 c_1(\lambda) \| p \|^2 + \| \lambda q \| M_4 \| p \| + \| \lambda r \| M_5 \| p \| + \| \lambda p \| M_3 \| s \| + \| x \| (M_1 c_1(\lambda) \| p \| \| r \| + M_2 c_2(\lambda) \| p \| \| q \| + M_3 c_3(\lambda) \| p \|^2)$$

où $\quad 0 < c_1, c_2, c_3 < \lambda$

$(p,q,r) = s \in P$ i.e. $\| s \|_\infty \leq 1$

M_1, M_2, M_3, M_4, M_5 sont les constantes de Lipschitz.

En passant de nouveau à la limite quand $\lambda \longrightarrow 0$, le second membre de l'inégalité tend vers 0 , et par suite, le premier membre tend aussi vers 0 .

Etant données les hypothèses de continuité sur f,g,h qui permettent de garantir le même signe entre 2 valeurs de ces fonctions (dans un certain voisinage), nous concluons

$\exists\ 0 \leq \tau_4 < 1$ tel que $\forall\ \varepsilon > 0\ \exists\ \bar{\lambda}_4 > 0$ tel que

$$(\ \divideontimes\ 4) \leq \tau_4 \frac{1}{\lambda} \| -u'(g(x+\lambda p) - g(x)) - v'(h(x+\lambda p)-h(x)) + x'(\nabla L(z+\lambda s) - \nabla L(z)) - \lambda q' g(x+\lambda p) - \lambda r' h(x+\lambda p) + \lambda p' \nabla L(z+\lambda s) \|_\infty + \varepsilon$$

$$\forall\ 0 < \lambda \leq \bar{\lambda}_4$$

5) En posant $\tau = \max \{ \tau_1\ \tau_2\ \tau_3\ \tau_4 \}$

$\lambda = \min \{ \bar{\lambda}_1\ \bar{\lambda}_2\ \bar{\lambda}_3\ \bar{\lambda}_4 \}$

on satisfait la condition c)

cqfd.

Si nous définissons la fonction

$$\theta(z) = e(z) - \min_{s \in P} \bar{e}(z,s)$$

le théorème II. §4. 3 nous dit que θ est une fonction d'optimalité de e(.) sur R^{n+m+1}.

Comme ce fut le cas aux chapitres II et III, nous considérons le problème $\min_{s \in P} \bar{e}(z,s)$ dont la solution s^* fournit une direction de descente pour la fonction e(z) (cfr II.§4.4)

En effet, soit s^* tel que $\min \bar{e}(z,s) = \bar{e}(z,s^*)$, alors ceci implique que $\theta(z) = e(z) - \bar{e}(z,s^*)$

Si de plus $\theta(z) > 0$ alors $\bar{e}(z,s^*) < e(z)$. Dès lors le corollaire II. §4. 4 nous assure que s^* est effectivement une direction de descente pour e(.).

Généralement, il est supposé que la minimisation de $\bar{e}(z,.)$ est plus aisée à résoudre que $\min_z e(z)$. Ce qui est précisément le cas ici car le choix de la norme $\| \|_\infty$ se justifie par le fait que le problème $\min_s \bar{e}(z,s)$ donne lieu à un problème de programmation linéaire.

§ 3. FORMULATION DU PROBLEME LINEAIRE.

Par définition de la norme $\|\ \|_\infty$, le problème $\min_{s \in P} \bar{e}(z,s)$ est encore équivalent à :

$$\text{minimiser} \|\ \nabla L(z) + \nabla^2 L(z)p + \nabla' g(x)q + \nabla' h(x)r,$$
$$\|s\| \leq 1 \quad (g(x) + \nabla g(x)p)_+,\ h(x) + \nabla h(x)p,\ (-u-q)_+,$$
$$(-u'g(x) - v'h(x) + x'\nabla L(z) - u'\nabla g(x)p -$$
$$v'\nabla h(x)p + \nabla' L(z)p - q'g(x) + q'\nabla g(x)x -$$
$$r'h(x) + r'\nabla h(x)x)_+ \|_\infty$$

Ce problème est du genre

$$\min_{|x| \leq 1} |x| \quad <==> \quad \min_{|x| \leq 1} \max \{x, -x\}$$

En posant $\max \{x, -x\} = \alpha$, le problème peut encore s'écrire

$$\begin{aligned} &\text{minimiser}_{x,\ \alpha} \quad \alpha \\ &\text{s.c.} \quad \alpha \geq x \\ &\qquad \alpha \geq -x \\ &\qquad -1 \leq x \leq 1 \end{aligned}$$

En effectuant le changement de variable $x' = x + 1$, nous obtenons :

$$\begin{aligned} &\text{minimiser}_{x',\ \alpha} \quad \alpha \\ &\text{s.c.} \quad x' - \alpha \leq 1 \\ &\qquad -x' - \alpha \leq -1 \\ &\qquad x' - 2 \leq 0 \\ &\qquad x' \geq 0 \\ &\qquad \alpha \geq 0 \end{aligned}$$

Si nous raisonnons de manière analogue pour $\bar{e}$, nous obtenons le problème linéaire en les inconnues α, p°, q°, r° :

$$\text{minimiser} \quad \alpha$$

$$\alpha,\ p^\circ,\ q^\circ, r^\circ$$

sous contraintes

$$\sum_{j=1}^{n} \frac{\partial \nabla_j L(z)}{\partial x_i} p_j^\circ + \sum_{j=1}^{m} \frac{\partial g_j(x)}{\partial x_i} q_j^\circ + \sum_{j=1}^{l} \frac{\partial h_j(x)}{\partial x_i} r_j^\circ - \alpha$$

$$\leq -\nabla_i L(z) + \sum_{j=1}^{n} \frac{\partial \nabla_j L(z)}{\partial x_i} + \sum_{j=1}^{m} \frac{\partial g_j(x)}{\partial x_i} + \sum_{j=1}^{l} \frac{\partial h_j(x)}{\partial x_i} \qquad i=1,..n$$

$$-\sum_{j=1}^{n} \frac{\partial \nabla_j L(z)}{\partial x_i} p_j^\circ - \sum_{j=1}^{m} \frac{\partial g_j(x)}{\partial x_i} q_j^\circ - \sum_{j=1}^{l} \frac{\partial h_j(x)}{\partial x_i} r_j^\circ - \alpha$$

$$\leq \nabla_i L(z) - \sum_{j=1}^{n} \frac{\partial \nabla_j L(z)}{\partial x_i} - \sum_{j=1}^{m} \frac{\partial g_j(x)}{\partial x_i} - \sum_{j=1}^{l} \frac{\partial h_j(x)}{\partial x_i} \qquad i=1,..n$$

$$\sum_{i=1}^{n} \frac{\partial g_j(x)}{\partial x_i} p_i^\circ - \alpha \leq -g_j(x) + \sum_{i=1}^{n} \frac{\partial g_j(x)}{\partial x_i} \qquad j=1,..m$$

$$\sum_{i=1}^{n} \frac{\partial h_k(x)}{\partial x_i} p_i^\circ - \alpha \leq -h_k(x) + \sum_{i=1}^{n} \frac{\partial h_k(x)}{\partial x_i} \qquad k=1,..l$$

$$-\sum_{i=1}^{n} \frac{\partial h_k(x)}{\partial x_i} p_i^\circ - \alpha \leq h_k(x) - \sum_{i=1}^{n} \frac{\partial h_k(x)}{\partial x_i} \qquad k=1,..l$$

$$-q_t^\circ - \alpha \leq u_t - l \qquad t=1,..m$$

$$(-u'\nabla g(x) - v'\nabla h(x) + \nabla' L(z) + x'\nabla^2 L(z))\ p^\circ + (-g'(x) + x'\nabla' g(x))\ q^\circ + (-h'(x) + x'\nabla' h(x))\ r^\circ - \alpha$$

$$\leq \quad u'g(x) + v'h(x) - x'\nabla L(z)$$
$$+ (-u'\nabla g(x) - v'\nabla h(x) + \nabla' L(z) + x'\nabla^2 L(z))\ 1_n$$
$$+ (-g'(x) + x'\nabla' g(x))\ 1_m$$
$$+ (-h'(x) + x'\nabla' h(x))\ 1_\ell$$

IV. §3. (1)

$$p_j^\circ - 2 \leq 0 \qquad j = 1,\ldots n$$
$$q_i^\circ - 2 \leq 0 \qquad i = 1,\ldots m$$
$$r_k^\circ - 2 \leq 0 \qquad k = 1,\ldots \ell$$
$$p_j^\circ \geq 0 \qquad j = 1,\ldots n$$
$$q_i^\circ \geq 0 \qquad i = 1,\ldots m$$
$$r_k^\circ \geq 0 \qquad k = 1,\ldots \ell$$
$$\alpha \geq 0$$

---

où $p_j^\circ = p_j + 1 \qquad j = 1,\ldots n$

$q_i^\circ = q_i + 1 \qquad i = 1,\ldots m$

$r_k^\circ = r_k + 1 \qquad k = 1,\ldots \ell$

Ayant défini e, $\bar{e}$, θ, nous pouvons décrire l'algorithme qui est une particularisation de l'algorithme III. § 1.

Soient f, g, h comme définies en IV. § 1.

Soit donné $0 < \varepsilon \leq 1$

Soit $P = \{ s \in R^{n+m+1}$ tel que $\| s \|_\infty \leq 1 \}$

<u>Initialisation</u> : $z^\circ \in R^{n+m+1}$

$k \longleftarrow 0$

aller à l'étape 1.

IV.§3.(2)

<u>Etape 1</u> : choix de $s^k \in P$ tel que

$$\bar{e}(z^k, s^k) = \min_{s \in P} \bar{e}(z^k, s)$$

aller à l'étape 2.

<u>Etape 2</u> : si $s^k = 0$ alors

si $e(z^k) = 0$ Stop : arrêt sur un pt de K.K.T.

sinon Stop : arrêt sur un pt stationnaire.

sinon aller à l'étape 3.

<u>Etape 3</u> : recherche linéaire le long de la direction s^k.

Soit λ^k le plus grand élément de

$\{1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots \}$ tel que

$$e(z^k) - e(z^k + \lambda^k s^k) \geq \varepsilon (\lambda^k)^2 \left(e(z^k) - \bar{e}(z^k, s^k) \right)$$

Poser $x^{k+1} = x^k + \lambda^k s^k$

$k \longleftarrow k + 1$

Retour à l'étape 1.

Si nous reprenons l'algorithme III. § 1., nous pouvons établir les correspondances suivantes :

$x \longleftrightarrow z$

$p \longleftrightarrow s$

$P(x) \longleftrightarrow P$

$\sigma : [0,\infty[\longrightarrow [0,\infty[\longleftrightarrow \mathrm{id}\ [0,\infty[\longrightarrow [0,\infty[$

strictement croissante et continue — strictement croissante et continue

$\theta(x) \longleftrightarrow \theta(z) = e(z) - \min_{s \in P} \bar{e}(z,s)$

$e(x^k) - \bar{e}(x^k,p^k) \geq \sigma(\theta(x^k)) \geq 0 \longleftrightarrow e(z^k) - e(z^k,s^k) = e(z^k) - \min_{s} \bar{e}(z^k,s)$

recherche linéaire $\longleftrightarrow$ pas d'Armijo.

Dès lors, nous pouvons appliquer les théorèmes établis au chapitre III à savoir III. § 2, III. § 4.

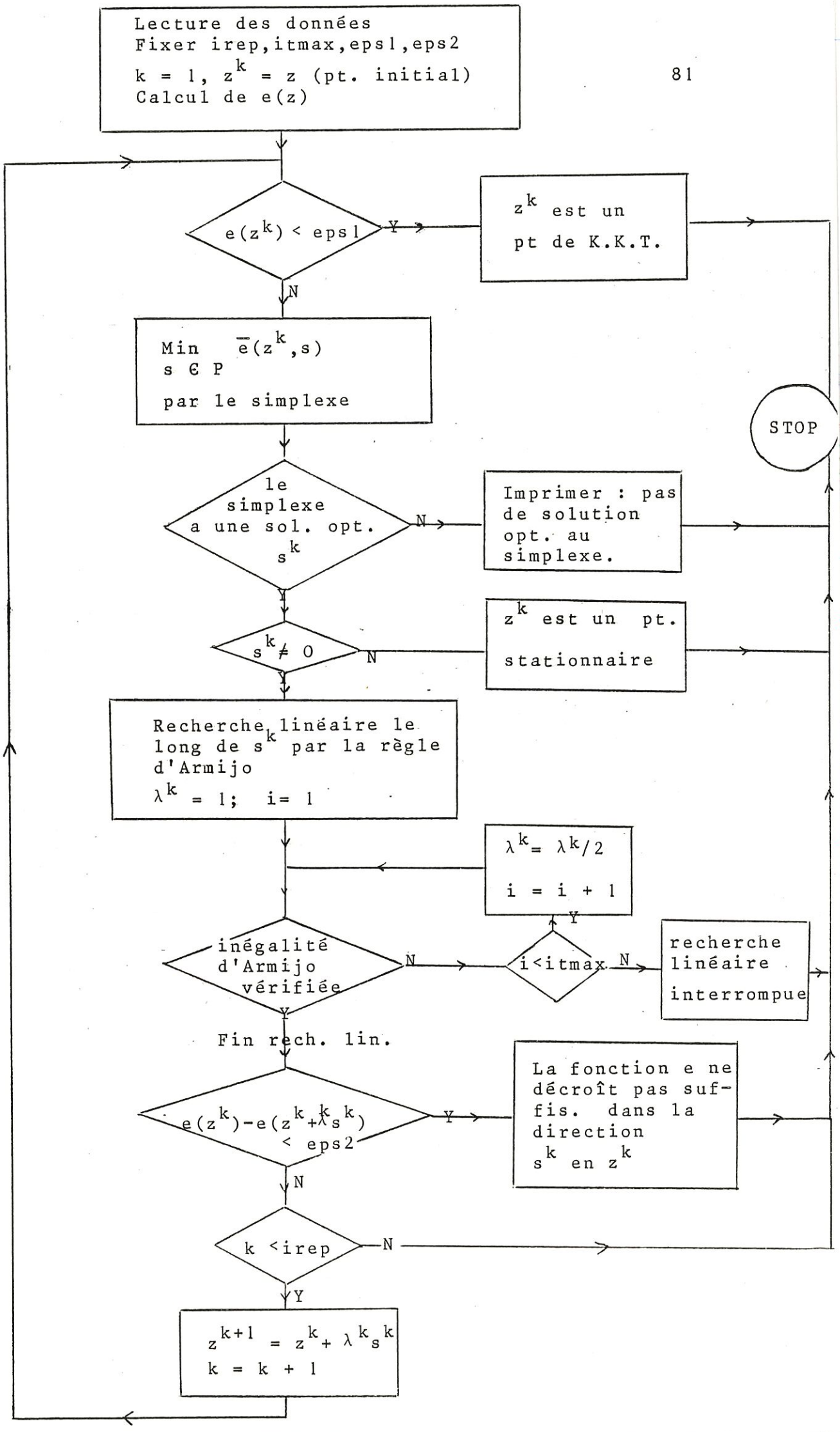
CHAPITRE V : PROGRAMMATION DE L'ALGORITHME.

Ce chapitre se situe au niveau de l'implémentation de l'algorithme IV.§3.(2). Nous donnons les principaux résultats des exemples qui ont été testés, quelques commentaires relatifs à la méthode, une comparaison entre l'aspect théorique et le point de vue numérique et enfin la conclusion. En annexe se trouvent le listing du programme ainsi que la table des variables.

Référence concernant les exemples testés : W.Hock (2)

§1. ORGANIGRAMME.

cfr page suivante.


Lecture des données
Fixer irep, itmax, eps1, eps2
k = 1, z^k = z (pt. initial)
Calcul de e(z)
$e(z^k) < eps1$
Y
z^k est un pt de K.K.T.
N
Min $\bar{e}(z^k, s)$
s ∈ P
par le simplexe
STOP
le simplexe a une sol. opt. s^k
N
Imprimer : pas de solution opt. au simplexe.
Y
$s^k \neq 0$
N
z^k est un pt. stationnaire
Y
Recherche linéaire le long de s^k par la règle d'Armijo
$\lambda^k = 1$; i = 1
$\lambda^k = \lambda^k/2$
i = i + 1
Y
inégalité d'Armijo vérifiée
N
i<itmax N
recherche linéaire interrompue
Y
Fin rech. lin.
$e(z^k) - e(z^k + \lambda^k s^k) < eps2$
Y
La fonction e ne décroît pas suffis. dans la direction s^k en z^k
N
k <irep
N
Y
$z^{k+1} = z^k + \lambda^k s^k$
k = k + 1

PROBLEME 1. W.Hock. (2) p.60

Nombre de variables : 3

Nombre de contraintes : 8

Minimiser $- x_1 x_2 x_3$

sous contraintes

$- 72 + x_1 + 2 x_2 + 2 x_3 \leq 0$

$- x_1 - 2 x_2 - 2 x_3 \leq 0$

$- x_1 \leq 0$

$- x_2 \leq 0$

$- x_3 \leq 0$

$x_1 - 42 \leq 0$

$x_2 - 42 \leq 0$

$x_3 - 42 \leq 0$

Point initial : $x_o = (10, 10, 10)$

$u_o = (0, 0, 0, 0, 0, 0, 0, 0)$

$f(x_o) = -1000$

Solution optimale : $x = (24, 12, 12)$

$f(x) = -3456$

```
DONNEES DE DEPART:
__________________

n=  3 m=  8 l=  0
ichang= 0

X[ 1]=  0.10000D+02
X[ 2]=  0.10000D+02
X[ 3]=  0.10000D+02
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00
U[ 4]=  0.00000D+00
U[ 5]=  0.00000D+00
U[ 6]=  0.00000D+00
U[ 7]=  0.00000D+00
U[ 8]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.10000D+03

 SOLUTION :
 __________

arret sur un PT de K.T

E(z)=  0.53991D-07

 nbre d'iterations  de l' algorithme :          17

 X[ 1] =-0.13502D-07
 X[ 2] = 0.13502D-07
 X[ 3] =-0.13502D-07
 U[ 1]= -0.13502D-07
 U[ 2]= -0.13499D-07
 U[ 3]= -0.13502D-07
 U[ 4]= -0.13502D-07
 U[ 5]=  0.40482D-07
 U[ 6]= -0.13502D-07
 U[ 7]= -0.13502D-07
 U[ 8]= -0.13502D-07


 Gx[ 1]= -0.72000D+02
 Gx[ 2]=  0.13502D-07
 Gx[ 3]=  0.13502D-07
 Gx[ 4]= -0.13502D-07
 Gx[ 5]=  0.13502D-07
 Gx[ 6]= -0.42000D+02
 Gx[ 7]= -0.42000D+02
 Gx[ 8]= -0.42000D+02

 Temps CPU :     7.05000
```

```
DONNEES DE DEPART:
__________________

n=  3 m=  8 l=  0
ichans= 1

X[ 1]=  0.10000D+02
X[ 2]=  0.10000D+02
X[ 3]=  0.10000D+02
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00
U[ 4]=  0.00000D+00
U[ 5]=  0.00000D+00
U[ 6]=  0.00000D+00
U[ 7]=  0.00000D+00
U[ 8]=  0.00000D+00

Eps1=  0.10000D-04    Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.10000D+03

 SOLUTION :
 __________

arret sur un PT.de K.T

E(z)=  0.31442D-17

 nbre d'iterations  de l' algorithme :          17

 X[ 1] =-0.54210D-18
 X[ 2] =-0.54210D-18
 X[ 3] =-0.75894D-18
 U[ 1]= -0.21684D-18
 U[ 2]=  0.21684D-18
 U[ 3]=  0.21684D-18
 U[ 4]=  0.10842D-18
 U[ 5]= -0.21684D-18
 U[ 6]= -0.32526D-18
 U[ 7]= -0.14095D-17
 U[ 8]= -0.18431D-17


 Gx[ 1]= -0.72000D+02
 Gx[ 2]=  0.31442D-17
 Gx[ 3]=  0.54210D-18
 Gx[ 4]=  0.54210D-18
 Gx[ 5]=  0.75894D-18
 Gx[ 6]= -0.42000D+02
 Gx[ 7]= -0.42000D+02
 Gx[ 8]= -0.42000D+02

 Temps CPU :     7.01300
```

PROBLEME 2. W.Hock. (2) p.66

Nombre de variables : 4

Nombre de contraintes : 3

$$\text{Min} \quad x_1^2 + x_2^2 + 2\,x_3^2 + x_4^2 - 5\,x_1 - 5\,x_2 - 21\,x_3 + 7\,x_4$$

sous contraintes

$$-8 + x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 \leq 0$$

$$-10 + x_1^2 + 2\,x_2^2 + x_3^2 + 2\,x_4^2 - x_1 - x_4 \leq 0$$

$$-5 + 2\,x_1^2 + x_2^2 + x_3^2 + 2\,x_1 - x_2 - x_4 \leq 0$$

Point initial : $x_o = (0, 0, 0, 0)$

$u_o = (0, 0, 0,)$

$f(x_o) = 0$

Solution optimale : $x = (0, 1, 2, -1)$

$f(x) = -44$

```
DONNEES DE DEPART:
------------------

n=  4 m=  3 l=  0
ichans= 0

X[ 1]=  0.00000D+00
X[ 2]=  0.00000D+00
X[ 3]=  0.00000D+00
X[ 4]=  0.00000D+00
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.250000+00
 E(z)= 0.210000+02

 SOLUTION :
 ---------

arret sur un PT de K,T

E(z)=  0.56537D-07

 nbre d'iterations  de l' algorithme :                10

 X[ 1] = 0.56265D-08
 X[ 2] = 0.10000D+01
 X[ 3] = 0.20000D+01
 X[ 4] =-0.10000D+01
 U[ 1]=  0.10000D+01
 U[ 2]=  0.99193D-07
 U[ 3]=  0.20000D+01


 Gx[ 1]=  0.12145D 07
 Gx[ 2]= -0.10000D+01
 Gx[ 3]=  0.12229D-07

 Temps CPU :     2.16300
```

```
DONNEES DE DEPART:
------------------

n=  4 m=  3 l=  0
ichans= 1

X[ 1]=  0.00000D+00
X[ 2]=  0.00000D+00
X[ 3]=  0.00000D+00
X[ 4]=  0.00000D+00
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.21000D+02

 SOLUTION :
 ---------

arret sur un PT de K.T

E(z)=  0.41870D-05

 nbre d'iterations  de l' algorithme :          13

 X[ 1] =-0.10879D-05
 X[ 2] = 0.10000D+01
 X[ 3] = 0.20000D+01
 X[ 4] =-0.10000D+01
 U[ 1]=  0.99999D+00
 U[ 2]= -0.43368D-18
 U[ 3]=  0.20000D+01


 Gx[ 1]=  0.24930D-05
 Gx[ 2]= -0.99999D+00
 Gx[ 3]= -0.19293D-05

 Temps CPU :     3.17300
```

```
DONNEES DE DEPART :
-------------------

n=  4 m=  3 l=  0
ichans= 0

X[ 1]=  0.00000D+00
X[ 2]=  0.00000D+00
X[ 3]=  0.00000D+00
X[ 4]=  0.00000D+00
U[ 1]=  0.50000D+01
U[ 2]=  0.40000D+01
U[ 3]=  0.30000D+01

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.95000D+02

 SOLUTION :
 ----------

arret sur un PT de K.T

E(z)=  0.11080D-05

 nbre d'iterations  de l' algorithme :          12

 X[ 1] = 0.11389D-06
 X[ 2] = 0.10000D+01
 X[ 3] = 0.20000D+01
 X[ 4] =-0.10000D+01
 U[ 1]=  0.10000D+01
 U[ 2]=  0.20690D-05
 U[ 3]=  0.20000D+01


 GxL 1]=  0.25375D-06
 GxL 2]= -0.10000D+01
 GxL 3]=  0.25378D-06

 Temps CPU :      2.48000
```

```
DONNEES DE DEPART:
__________________

n=  4 m=  3 l=  0
ichang= 1

X[ 1]=  0.00000D+00
X[ 2]=  0.00000D+00
X[ 3]=  0.00000D+00
X[ 4]=  0.00000D+00
U[ 1]=  0.50000D+01
U[ 2]=  0.40000D+01
U[ 3]=  0.30000D+01

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.95000D+02

 SOLUTION :
 __________

arret sur un PT de K.T

E(z)=  0.33267D-06

 nbre d'iterations  de l' algorithme :                10

 X[ 1] =-0.86787D-07
 X[ 2] = 0.10000D+01
 X[ 3] = 0.20000D+01
 X[ 4] =-0.10000D+01
 U[ 1]=  0.10000D+01
 U[ 2]=  0.65052D-18
 U[ 3]=  0.20000D+01


 Gx[ 1]=  0.57698D-07
 Gx[ 2]= -0.10000D+01
 Gx[ 3]= -0.18799D-06

 Temps CPU :     2.22800
```

PROBLEME 3. W. Hock. (2) p.85

Nombre de variables : 3

Nombre de contraintes : 6

$$\text{Min} \quad 1000 - x_1^2 - 2\,x_2^2 - x_3^2 - x_1\,x_2 - x_1\,x_3$$

sous contraintes

$$-x_1 \leq 0$$

$$-x_2 \leq 0$$

$$-x_3 \leq 0$$

$$8\,x_1 + 14\,x_2 + 7\,x_3 - 56 = 0$$

$$x_1^2 + x_2^2 + x_3^2 - 25 = 0$$

Point initial : $x_o = (2, 2, 2)$

$u_o = (0, 0, 0)$

$v_o = (0, 0)$

Solution optimale : $x = (3.5121184, 0.21698817, 3.5521740)$

$f(x) = 961.7151721$

```
DONNEES DE DEPART:
------------------

n=  3 m=  3 l=  2
ichans= 1

XE 1]=  0.20000D+01
XE 2]=  0.20000D+01
XE 3]=  0.20000D+01
UE 1]=  0.00000D+00
UE 2]=  0.00000D+00
UE 3]=  0.00000D+00
VE 1]=  0.00000D+00
VE 2]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.13000D+02

 SOLUTION :
 ----------

arret sur un PT de K.T

E(z)=  0.37557D-05

 nbre d"iterations  de l" algorithme :              6

 XE 1] = 0.35121D+01
 XE 2] = 0.21699D+00
 XE 3] = 0.35522D+01
 UE 1]= -0.43368D-18
 UE 2]= -0.86736D-18
 UE 3]= -0.21684D-18
 VE 1]=  0.27494D+00
 VE 2]=  0.12235D+01


 GxE 1]= -0.35121D+01
 GxE 2]= -0.21699D+00
 GxE 3]= -0.35522D+01
 HxE 1]=  0.13878D-16
 HxE 2]=  0.37557D-05

 Temps CPU :      1.51600
```

```
DONNEES DE DEPART:
------------------

n=  3 m=  3 l=  2
ichans= 0

X[ 1]=  0.20000D+01
X[ 2]=  0.20000D+01
X[ 3]=  0.20000D+01
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00
V[ 1]=  0.00000D+00
V[ 2]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.13000D+02

 SOLUTION :
 ----------

arret sur un PT de K.T

E(z)=  0.17550D-07

 nbre d'iterations  de l' algorithme :              7

 X[ 1] = 0.35121D+01
 X[ 2] = 0.21699D+00
 X[ 3] = 0.35522D+01
 U[ 1]= -0.35980D-10
 U[ 2]= -0.35980D-10
 U[ 3]= -0.35980D-10
 V[ 1]=  0.27494D+00
 V[ 2]=  0.12235D+01


 Gx[ 1]= -0.35121D+01
 Gx[ 2]= -0.21699D+00
 Gx[ 3]= -0.35522D+01
 Hx[ 1]= -0.35980D-10
 Hx[ 2]=  0.17550D-07

 Temps CPU :     1.68400
```

```
DONNEES DE DEPART:
------------------

n=  3 m=  3 l=  2
ichans= 1

X[ 1]=  0.20000D+01
X[ 2]=  0.20000D+01
X[ 3]=  0.20000D+01
U[ 1]=  0.10000D+02
U[ 2]=  0.10000D+02
U[ 3]=  0.10000D+02
V[ 1]=  0.00000D+00
V[ 2]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.60000D+02

 SOLUTION :
 ----------

 La descente n'est pas suffisante.
 Ez ne decroit plus. Pas moyen d'obtenir un
 pt de K.T


 nbre d'iterations  de l' algorithme :                12

 X[ 1] =-0.71888D+00
 X[ 2] = 0.48216D+01
 X[ 3] =-0.71888D+00
 U[ 1]=  0.40000D+01
 U[ 2]= -0.71888D+00
 U[ 3]=  0.79789D+01
 V[ 1]=  0.84264D+00
 V[ 2]=  0.55300D+00


 Gx[ 1]=  0.71888D+00
 Gx[ 2]= -0.48216D+01
 Gx[ 3]=  0.71888D+00
 Hx[ 1]=  0.71888D+00
 Hx[ 2]= -0.71888D+00

 Temps CPU :     3.04700
```

PROBLEME 4. W. Hock. p.74 (2)

Nombre de variables : 5

Nombre de contraintes : 3

$$\text{Min } (x_1 - x_2)^2 + (x_2 + x_3 - 2)^2 + (x_4 - 1)^2 + (x_5 - 1)^2$$

sous contraintes

$$x_1 + 3\,x_2 - 4 = 0$$

$$x_3 + x_4 - 2\,x_5 = 0$$

$$x_2 - x_5 = 0$$

Point initial : $x_o = (2.5,\ 0.5,\ 2,\ -1,\ 0.5)$

$v_o = (0,\ 0,\ 0)$

Solution optimale : $x = (1,\ 1,\ 1,\ 1,\ 1)$

$f(x) = 0$

```
Recherche d'un pt de K.K.T du pb
--------------------------------

Min (X(1)-X(2))**2 + (X(2)+X(3)-2)**2
    + (X(4)-1)**2 + (X(5)-1)**2
Sous contraintes
    X(1) + 3X(2) - 4 =0
    X(3) + X(4) - 2X(5) =0
    X(2) - X(5) =0

DONNEES DE DEPART:
------------------

n=  5 m=  0 l=  3 ichang= 1
X[ 1]=  0.25000D+01
X[ 2]=  0.50000D+00
X[ 3]=  0.20000D+01
X[ 4]= -0.10000D+01
X[ 5]=  0.50000D+00
V[ 1]=  0.00000D+00
V[ 2]=  0.00000D+00



V[ 3]=  0.00000D+00
Eps1=  0.10000D-04   Eps2=  0.10000D-09
alpha=  0.10000D+01
 E(z)= 0.40000D+01

 SOLUTION :
 ----------

arret sur un PT de K.T

Ez=  0.78137D-05

 nbre d'iterations  de l'' algorithme :                21

 X[ 1] = 0.10000D+01
 X[ 2] = 0.10000D+01
 X[ 3] = 0.99999D+00
 X[ 4] = 0.99999D+00
 X[ 5] = 0.99999D+00
 V[ 1]=  0.14630D-04
 V[ 2]=  0.16627D-04
 V[ 3]= -0.43535D-04


 HxL 1]=  0.91912D-06
 HxL 2]= -0.65605D-05
 HxL 3]=  0.78137D-05

 Temps CPU :      4.98100
```

```
Recherche d'un pt de K.K.T du pb
--------------------------------

Min (X(1)-X(2))**2 + (X(2)+X(3)-2)**2
    + (X(4)-1)**2 + (X(5)-1)**2
Sous contraintes
    X(1) + 3X(2) - 4 =0
    X(3) + X(4) - 2X(5) =0
    X(2) - X(5) =0

DONNEES DE DEPART:
------------------
n=  5 m=  0 l=  3 ichain= 0
X[ 1]=  0.250000D+01
X[ 2]=  0.500000D+00
X[ 3]=  0.200000D+01
X[ 4]= -0.100000D+01
X[ 5]=  0.500000D+00
V[ 1]=  0.000000D+00
V[ 2]=  0.000000D+00
V[ 3]=  0.000000D+00
Eps1=  0.100000D-04    Eps2=  0.10000D-09
alpha=  0.250000D+00
 E(z)= 0.140000D+02
 SOLUTION
 --------

arret sur un Pf de K.T

Ez=  0.563790D-17

 nbre d'iterations  de l' algorithme :          3

 X[ 1] = 0.100000D+01
 X[ 2] = 0.100000D+01
 X[ 3] = 0.100000D+01
 X[ 4] = 0.100000D+01
 X[ 5] = 0.100000D+01
 V[ 1]=  0.5637[illegible]D-17
 V[ 2]=  0.91[illegible]D-17
 V[ 3]= -0.19[illegible]

 Hx[ 1]= -0.[illegible]
 Hx[ 2]= -0.[illegible]
 Hx[ 3]=  0.[illegible]

 Temps CPU :      0.[illegible]
```

PROBLEME 5. W. Hock. (2) p. 111

Nombre de variables : 7

Nombre de contraintes : 4

$$\text{Min } (x_1 - 10)^2 + 5\,(x_2 - 12)^2 + x_3^4 + 3\,(x_4 - 11)^2 + 10\,x_5^6 + 7\,x_6^2 + x_7^4 - 4\,x_6\,x_7 - 10\,x_6 - 8\,x_7$$

sous contraintes

$$-127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0$$

$$-282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0$$

$$-196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0$$

$$+4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0$$

Point initial : $x_o = (1, 2, 0, 4, 0, 1, 1)$

$u_o = (0,0,0,0)$

$f(x_o) = 714$

Solution optimale : $x = (2.330499, 1.951372, -0.4775414, 4.365726, -0.6244870, 1.038131, 1.594227)$

$f(x) = 680.6300573$

```
DONNEES DE DEPART:
------------------

n=  7 m=  4 l=  0
ichans= 1

X[ 1]=  0.10000D+01
X[ 2]=  0.20000D+01
X[ 3]=  0.00000D+00
X[ 4]=  0.40000D+01
X[ 5]=  0.00000D+00
X[ 6]=  0.10000D+01
X[ 7]=  0.10000D+01
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00
U[ 4]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.10000D+03

 SOLUTION :
 ----------

arret sur un PT de K.T

E(z)=  0.45186D-05

 nbre d'iterations  de l' algorithme :          19

 X[ 1] = 0.23305D+01
 X[ 2] = 0.19514D+01
 X[ 3] =-0.47754D+00
 X[ 4] = 0.43657D+01
 X[ 5] =-0.62449D+00
 X[ 6] = 0.10381D+01
 X[ 7] = 0.15942D+01
 U[ 1]=  0.11397D+01
 U[ 2]=  0.00000D+00
 U[ 3]=  0.21684D-18
 U[ 4]=  0.36861D+00


 GxL[ 1]=  0.37554D-12
 GxL[ 2]= -0.25256D+03
 GxL[ 3]= -0.14488D+03
 GxL[ 4]=  0.28166D-13

 Temps CPU :     8.13400
```

```
DONNEES DE DEPART:
------------------

n=  7 m=  4 l=  0
ichang= 0

X[ 1]=  0.10000D+01
X[ 2]=  0.20000D+01
X[ 3]=  0.00000D+00
X[ 4]=  0.40000D+01
X[ 5]=  0.00000D+00
X[ 6]=  0.10000D+01
X[ 7]=  0.10000D+01
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00
U[ 4]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.10000D+03

 SOLUTION :
 ----------

 reche.lin.interrompue car itmax est atteint
sans avoir trouve le pas Lamda.

 E(z)=  0.29415D+00
Ezs=  0.00000D+00

 nbre d'iterations  de l' algorithme :          6

 X[ 1] = 0.23064D+01
 X[ 2] = 0.19450D+01
 X[ 3] =-0.46599D+00
 X[ 4] = 0.43407D+01
 X[ 5] =-0.62639D+00
 X[ 6] = 0.10346D+01
 X[ 7] = 0.15881D+01
 U[ 1]=  0.11479D+01
 U[ 2]= -0.42013D-18
 U[ 3]= -0.13553D-19
 U[ 4]=  0.36260D+00


 GxL[ 1]= -0.16577D+01
 GxL[ 2]= -0.25288D+03
 GxL[ 3]= -0.14545D+03
 GxL[ 4]= -0.25807D+00

 Temps CPU :      4.06600
```

PROBLEME 6. Colville No. 1 W. Hock. (2) p. 105

Nombre de variables : 5

Nombre de contraintes : 15

$$\text{Min} \sum_{j=1}^{5} e_j x_j + \sum_{i=1}^{5} \sum_{j=1}^{5} c_{ij} x_i x_j + \sum_{j=1}^{5} d_j x_j^3$$

sous contraintes

$$\sum_{j=1}^{5} a_{ij} x_j - b_i \geq 0 \qquad i = 1, \ldots 10$$

$$0 \leq x_i \qquad i = 1, \ldots 5$$

j	1	2	3	4	5
e_j	-15	-27	-36	-18	-12
c_{1j}	30	-20	-10	32	-10
c_{2j}	-20	39	-6	-31	32
c_{3j}	-10	-6	10	-6	-10
c_{4j}	32	-31	-6	39	-20
c_{5j}	-10	32	-10	-20	30
d_j	4	8	10	6	2
a_{1j}	-16	2	0	1	0
a_{2j}	0	-2	0	4	2
a_{3j}	-3.5	0	2	0	0
a_{4j}	0	-2	0	-4	-1
a_{5j}	0	-9	-2	1	-2.8
a_{6j}	2	0	-4	0	0
a_{7j}	-1	-1	-1	-1	-1
a_{8j}	-1	-2	-3	-2	-1
a_{9j}	1	2	3	4	5
a_{10j}	1	1	1	1	1
b_j	-40	-2	-0.25	-4	-4
b_{5+j}	-1	-40	-60	5	1

Point initial : x_o = (0, 0, 0, 0, 1)

u_o = (0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)

$f(x_o)$ = 20

Solution optimale : x = (0.3, 0.33346761, 0.4, 0.42831010,

0.22396487)

f(x) = -32.34867897

```
DONNEES DE DEPART:
_____________________
n=  5 m= 15 l=  0 ichang= 0
X[ 1]=  0.00000D+00
X[ 2]=  0.00000D+00
X[ 3]=  0.00000D+00
X[ 4]=  0.00000D+00
X[ 5]=  0.10000D+01
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00
U[ 4]=  0.00000D+00
U[ 5]=  0.00000D+00
U[ 6]=  0.00000D+00
U[ 7]=  0.00000D+00
U[ 8]=  0.00000D+00
U[ 9]=  0.00000D+00
U[10]=  0.00000D+00
U[11]=  0.00000D+00
U[12]=  0.00000D+00
U[13]=  0.00000D+00
U[14]=  0.00000D+00
U[15]=  0.00000D+00
Eps1=  0.10000D-04   Eps2=  0.10000D-09
alpha=  0.25000D+00
 E(z)= 0.58000D+02
```

```
 SOLUTION :
 ----------

arret sur un PT de K.T

Ez=  0.92144D-05

 nbre d"iterations  de l" algorithme :                32

 X[ 1] = 0.30000D+00
 X[ 2] = 0.33347D+00
 X[ 3] = 0.40000D+00
 X[ 4] = 0.42831D+00
 X[ 5] = 0.22396D+00
 U[ 1]=  0.25978D-04
 U[ 2]= -0.92144D-05
 U[ 3]=  0.51739D+01
 U[ 4]= -0.92144D-05
 U[ 5]=  0.30611D+01
 U[ 6]=  0.11839D+02
 U[ 7]= -0.92144D-05
 U[ 8]= -0.92144D-05
 U[ 9]=  0.10390D+00
 U[10]= -0.92144D-05
 U[11]= -0.92144D-05




 U[12]= -0.92144D-05
 U[13]= -0.92144D-05
 U[14]= -0.92144D-05
 U[15]= -0.92144D-05


 Gx[ 1]= -0.36295D+02
 Gx[ 2]= -0.34942D+01
 Gx[ 3]=  0.10842D-17
 Gx[ 4]= -0.13959D+01
 Gx[ 5]=  0.99747D-17
 Gx[ 6]=  0.00000D+00
 Gx[ 7]= -0.38314D+02
 Gx[ 8]= -0.56752D+02
 Gx[ 9]= -0.73726D-17
 Gx[10]= -0.68574D+00
 Gx[11]= -0.30000D+00
 Gx[12]= -0.33347D+00
 Gx[13]= -0.40000D+00
 Gx[14]= -0.42831D+00
 Gx[15]= -0.22396D+00

 Temps CPU :     65.36200
```

```
DONNEES DE DEPART:
------------------

n=  5 m= 15 l=  0
ichang= 0

X[ 1]=  0.50000D+01
X[ 2]=  0.50000D+01
X[ 3]=  0.50000D+01
X[ 4]=  0.50000D+01
X[ 5]=  0.50000D+01
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00
U[ 4]=  0.00000D+00
U[ 5]=  0.00000D+00
U[ 6]=  0.00000D+00
U[ 7]=  0.00000D+00
U[ 8]=  0.00000D+00
U[ 9]=  0.00000D+00
U[10]=  0.00000D+00
U[11]=  0.00000D+00
U[12]=  0.00000D+00
U[13]=  0.00000D+00
U[14]=  0.00000D+00
U[15]=  0.00000D+00

Eps1=  0.10000D-04    Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(x)= 0.13210D+05
```

```
 SOLUTION :
 ----------

arret sur un PT de K.T

E(z)=  0.19268D-05

 nbre d"iterations  de l" algorithme :                73

 X[ 1] = 0.30000D+00
 X[ 2] = 0.33347D+00
 X[ 3] = 0.40000D+00
 X[ 4] = 0.42831D+00
 X[ 5] = 0.22396D+00
 U[ 1]=  0.54324D-05
 U[ 2]= -0.19268D-05
 U[ 3]=  0.51740D+01
 U[ 4]= -0.19268D-05
 U[ 5]=  0.30611D+01
 U[ 6]=  0.11840D+02
 U[ 7]= -0.19268D-05
 U[ 8]= -0.19268D-05
 U[ 9]=  0.10390D+00
 U[10]= -0.19268D-05
 U[11]= -0.19268D-05
 U[12]= -0.19268D-05
 U[13]= -0.19268D-05
 U[14]= -0.19268D-05



 U[15]= -0.19268D-05


 G*[ 1]= -0.36295D+02
 G*[ 2]= -0.34942D+01
 G*[ 3]=  0.15179D-17
 G*[ 4]= -0.13959D+01
 G*[ 5]= -0.32526D-17
 G*[ 6]= -0.21684D-17
 G*[ 7]= -0.38314D+02
 G*[ 8]= -0.56752D+02
 G*[ 9]=  0.17347D-17
 G*[10]= -0.68574D+00
 G*[11]= -0.30000D+00
 G*[12]= -0.33347D+00
 G*[13]= -0.40000D+00
 G*[14]= -0.42831D+00
 G*[15]= -0.22396D+00

 Temps CPU :    159.00600
```

```
DONNEES DE DEPART:
------------------

n=  5 m= 15 l=  0
ichang= 0

X[ 1]=  0.50000D+01
X[ 2]=  0.50000D+01
X[ 3]=  0.50000D+01
X[ 4]=  0.50000D+01
X[ 5]=  0.50000D+01
U[ 1]=  0.00000D+00
U[ 2]=  0.00000D+00
U[ 3]=  0.00000D+00
U[ 4]=  0.00000D+00
U[ 5]=  0.00000D+00
U[ 6]=  0.00000D+00
U[ 7]=  0.00000D+00
U[ 8]=  0.00000D+00
U[ 9]=  0.00000D+00
U[10]=  0.00000D+00
U[11]=  0.00000D+00
U[12]=  0.00000D+00
U[13]=  0.00000D+00
U[14]=  0.00000D+00
U[15]=  0.00000D+00

Eps1=  0.10000D-04   Eps2=  0.10000D-09

alpha=  0.25000D+00
 E(z)= 0.13210D+05
```

SOLUTION :

```
arret sur un PT de K.T

E(z)=  0.19268D-05

 nbre d"iterations  de l" algorithme :                73

 X[ 1] = 0.30000D+00
 X[ 2] = 0.33347D+00
 X[ 3] = 0.40000D+00
 X[ 4] = 0.42831D+00
 X[ 5] = 0.22396D+00
 U[ 1]=  0.54324D-05
 U[ 2]= -0.19268D-05
 U[ 3]=  0.51740D+01
 U[ 4]= -0.19268D-05
 U[ 5]=  0.30611D+01
 U[ 6]=  0.11840D+02
 U[ 7]= -0.19268D-05
 U[ 8]= -0.19268D-05
 U[ 9]=  0.10390D+00
 U[10]= -0.19268D-05
 U[11]= -0.19268D-05
 U[12]= -0.19268D-05
 U[13]= -0.19268D-05
 U[14]= -0.19268D-05



 U[15]= -0.19268D-05


 Gx[ 1]= -0.36295D+02
 Gx[ 2]= -0.34942D+01
 Gx[ 3]=  0.15179D-17
 Gx[ 4]= -0.13959D+01
 Gx[ 5]= -0.32526D-17
 Gx[ 6]= -0.21684D-17
 Gx[ 7]= -0.38314D+02
 Gx[ 8]= -0.56752D+02
 Gx[ 9]=  0.17347D-17
 Gx[10]= -0.68574D+00
 Gx[11]= -0.30000D+00
 Gx[12]= -0.33347D+00
 Gx[13]= -0.40000D+00
 Gx[14]= -0.42831D+00
 Gx[15]= -0.22396D+00

 Temps CPU :    159.00600
```

A N N E X E

```
c      ***********************************************************
       SUBROUTINE MATSIM (A,B,C,D,LAPL,JBG,JBH,n,m,i,ndim)
c      ***********************************************************
c
c      formation de la matrice A du simplex, et du vecteur
c      independant B
c
c      Declarations
c      ------------
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       DOUBLE PRECISION A(ndim,1),B(1),C(1),D(1),LAPL(ndim,1),
      1                 JBH(ndim,1),JBG(ndim,1)
c
c      Corps
c      -----
       k3=m+n+1+1
       DO 10 i=1,3*(m+n+1)+2
             DO 20 J=1,4*(m+n+1)+2
                   A(i,J)=0.
20           continue
10     continue
       A(1,k3)=1.
       B(1)=0.
c
       DO 30 i=1,n
             sisne=1.
             If ( C(i).LT.0. ) sisne=-1.
             B(i+1)=C(i)*sisne
             B(i+n+1)=B(i+1)
c
             DO 40 J=1,n
                   A(i+1,J)=LAPL(i,J)*sisne
                   A(i+1+n,J)=A(i+1,J)
40           continue
c
             DO 50 J=n+1,n+m
                   A(i+1,J)=JBG(J-n,i)*sisne
                   A(i+1+n,J)=A(i+1,J)
50           continue
c
             DO 60 J=n+m+1,n+m+1
                   A(i+1,J)=JBH(J-n-m,i)*sisne
                   A(i+1+n,J)=A(i+1,J)
60           continue
c
             A(i+1,k3)= -sisne
             A(i+1+n,k3)= sisne
             A(i+1,k3+i)= sisne
             A(i+1+n,k3+i+n)= -sisne
c
30     continue
c
       k1= 3*n+m+1+2
       k2= 2*n+2
c
       DO 70 i=n+1,n+m
             sisne=1.




             If ( C(i).LT.0. ) sisne=-1.
             B(k2)=C(i)*sisne
c
             DO 80 J=1,n
                   A(k2,J)=JBG(i-n,J)*sisne
```

```
80              continue
                A(k2,k3)= -sisne
                A(k2,k1)= sisne
c
                k1=k1+1
                k2=k2+1
c
70      continue
c
        DO 90 i=n+m+1,n+m+1
                sisne=1.
                If ( C(i).LT.0. ) sisne= -1.
                B(k2)=C(i)*sisne
                B(k2+1)=B(k2)
c
                DO 100 J=1,n
                        A(k2,J)=JBH(i-n-m,J)*sisne
                        A(k2+1,J)=A(k2,J)
100             continue
c
                A(k2,k3)= -sisne
                A(k2+1,k3)= sisne
                A(k2,k1)= sisne
                A(k2+1,k1+1)= -sisne
                k1=k1+1
                k2=k2+1
c
90      continue
c
        k1=k1+1
        k2=k2+1
        k=n+1
c
        DO 110 i=k3,n+2*m+1
                sisne=1.
                If ( C(i).LT.0. ) sisne= -1.
                B(k2)=C(i)*sisne
                A(k2,k)= -sisne
                A(k2,k3)= -sisne
                A(k2,k1)= sisne
                k1=k1+1
                k2=k2+1
                k=k+1
c
110     continue
c
        sisne=1.
        If ( C(n+2*m+1+1).LT.0. ) sisne= -1.
        B(k2)=C(n+2*m+1+1)*sisne
c
        DO 120 i=1,k3
                A(k2,i)=D(i)*sisne
120     continue
c
        A(k2,k1)=sisne
        k2=k2+1




        k1=k1+1
c
        DO 130 i=1,n+m+1
                B(k2)=2.
                A(k2,i)=1.
                A(k2,k1)=1.
                k2=k2+1
                k1=k1+1
130     continue
```

```
      RETURN
      END
c
c
c
c     ***********************************************************
      SUBROUTINE LECTUR(X,U,V,n,m,l,ndim,eps1,eps2,itmax,
     1                  alpha, irep,ires,ichans)
c     ***********************************************************
c
c     lecture des donnees du probleme de depart.
c
c     Variables
c     ---------
c         eps1 : y=0 a eps1 pres.
c         eps2 : x-y=0 a eps2 pres.
c
c     Declarations
c     ------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION X(n),U(m),V(l)
c
c     Corps
c     -----
      Write(5,9000)
 9000 Format(1x,'BonJour!',//x,'Ce programme consiste',1x,
     1 'a determiner un point de K.K.T. du probleme suivant:',//x,
     2 '              Minimiser f(x) ',/x,
     3 '              s.c.   g(x) <= 0 ',/x,
     4 '                     h(x) = 0 ',//x,'Pour de plus amples ',
     5 'renseignements, consultez le fichier INFO.HLP.')
      Write(5,8000)
 8000 format(1x,' No du fichier de donnees? ',$)
      read(5,*)ilect
c
      If (ilect.NE.5) goto 60
      Write (5,9010)
 9010 Format(/x,'Nombre de variables : ',$)
      Read (5,*)n
      Write (5,9020)
 9020 Format(/x,'Nombre de contraintes d"inegalite : ',$)
      Read (5,*)m
      Write (5,9030)
 9030 Format (/x,'Nombre de contraintes d"egalite : ',$)
      Read (5,*)l
      Write (5,9040)
 9040 Format (/x,'Composantes du vecteur initial X',/x,
     1 '( 1 composante par ligne )')
      Do 100 i=1,n
            Read (5,*)X(i)
 100  continue



      Write (5,9050)
 9050 Format (/x,'Composantes du vecteur multiplicateur de',1x,
     1 'Lagrange associe',/x,'aux contraintes d"inegalite :',/x,
     2 '( 1 composante par ligne )')
      Do 110 i=1,m
            Read (5,*)U(i)
 110  continue
      Write (5,9060)
 9060 Format (/x,'Composantes du vecteur multiplicateur de',1x,
     1 'Lagrange associe',/x,'aux contraintes d"egalite :',/x,
     2 '( 1 composante par ligne )')
      Do 120 i=1,l
            Read (5,*)V(i)
```

```
120     continue
        Write (5,9070)
9070    Format (/x,'Precision du zero Eps1 : ',$)
        Read (5,*)eps1
        Write (5,9080)
9080    Format (/x,'Precision du zero Eps2 : ',$)
        Read (5,*)eps2
        Write (5,9090)
9090    Format (/x,'Parametre "alpha" de la regle d"Armijo :',$)
        Read (5,*)alpha
        Write (5,9005)
9005    Format (/x,'Nombre d"iterations total requis : ',$)
        Read (5,*)irep
        Write (5,9015)
9015    Format (/x,'Nombre d"iterations maximal dans la ',
        1 'recherche lineaire : ',$)
        Read (5,*)itmax
        Write (5,9025)
9025    Format(/x,'Numero du fichier de resultats : ',$)
        Read (5,*)ires
        Write (5,9035)
9035    Format(/x,'Numero de l"option choisie ?',/x,
        1 ' 1, si la methode utilise les conditions de ',
        2 'K.K.T. habituelles',/x,
        3 '0, sinon ',/x,'Votre numero choisi : ',$)
        Return
c
60      read(ilect,*)n,m,l
        If ((4*(n+m+l)+2).GT.ndim) goto 50
        Do 10 i=1,n
              read(ilect,*)X(i)
10      continue
c
        Do 20 i=1,m
              read(ilect,*)U(i)
20      continue
c
        Do 30 i=1,l
              read(ilect,*)V(i)
30      continue
c
        read(ilect,*)eps1
        read(ilect,*)eps2
c       (alpha=precis. de l"approximant  ds reclin)
        read(ilect,*)alpha
c
c       (irep=nbre d"iter. max. ds P.P.)
        read(ilect,*)irep
c
c       (itmax=nbre d"iter.max. ds reclin)
        read(ilect,*)itmax
c
        read(ilect,*)ires
c
        read(ilect,*)ichans
        RETURN
c
50      Write(5,8100)
8100    format(1x,' Erreur de dimensionnement: la condition',/x,
        1          ' 4*(n+m+l)+2 <=ndim n"est pas verifiee.')
        STOP
        END
c
c
```

```
c
c       ************
        PROGRAM PRINCI
c       ************
c
c
c       Pour obtenir des informations concernant le programme,
c       consulter le fichier INFO.HLP
c       Pour la creation d'un fichier de donnees,consulter
c       le fichier FICH.HLP
c
c       Declarations
c       ------------
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DOUBLE PRECISION X(20),U(20),V(20),Gx(20),Hx(20),LAGR(20),
     1                   DF(20),P(20),Q(20),R(20),TOL(4),
     1                   ERR(8),S(100),T(100),Y(100),
     1                   B(100),C(100),D(25),PQR(100),JBG(100,20),
     1                   JBH(100,20),LAPL(100,20),LAPLF(100,20),
     1                   LAPLG(100,20),LAPLH(100,20),A(100,100),
     1                   E(100,100),
     1                   Time1,Time2
        DIMENSION INFIX(8),KOUT(7),KB(100),JH(100)
c
c       Corps
c       -----
        ndim=100
c
        call LECTUR(X,U,V,n,m,l,ndim,eps1,eps2,itmax,alpha,
     1              ire,ires,ichans)
        call ECRIT(ires)
        call SECOND(Time1)
        call DONNEE(ires,n,m,l,ichans,X,U,V,eps1,eps2,alpha)
c
c       (initialisations)
c
        INFIX(1)=4
        INFIX(2)=4*(n+m+1)+2
        INFIX(3)=100
        INFIX(4)=3*(n+m+1)+2
        INFIX(5)=2
        INFIX(6)=1
        INFIX(7)=100
        INFIX(8)=0
        PRM=0.
        TOL(1)=1.D-5
        TOL(2)=1.D-5
        TOL(3)=-1.D-3
        TOL(4)=1.D-10
c
        call CINEG(X,Gx,n,m)
        call CEGAL(X,Hx,n,l)
        call GRADF(X,DF,n)
        call JACOG(X,JBG,ndim,n)
        call JACOH(X,JBH,ndim,n)
        call LAGRAN(DF,JBG,JBH,U,V,n,m,l,ndim,LAGR)
        call FCTOBJ(LAGR,Hx,Gx,X,U,V,n,m,l,Ez,ichans)
c
        Write(ires,8570)Ez
8570    format(1x,' E(z)=',D12.5)
c
        ideb=1
        itest=3
c
c       (corps)
c
```

```
25      Do 150 Jk=ideb,ifin
               nbite=Jk
c
               If (Ez.LT.eps1) goto 500
c
               call LAPLAG(X,LAPLG,ndim,n)
               call LAPLAH(X,LAPLH,ndim,n)
               call LAPLAF(X,LAPLF,ndim,n)
               call LAPLAC(LAPLF,LAPLG,LAPLH,
     1                         U,V,n,m,l,ndim,LAPL)
c
c      formation de matr. du simplex.
c
               If (ichang.eq.1) goto 576
               call TERIND(LAGR,LAPL,JBG,JBH,Gx,
     1                         Hx,X,U,V,C,n,m,l,ndim)
               call AUXIL(JBG,LAPL,JBH,D,X,U,V,
     1                         ndim,LAGR,Hx,Gx,n,m,l)
               goto 577
576            call TERIN2(LAGR,LAPL,JBG,JBH,Gx,
     1                         Hx,X,U,V,C,n,m,l,ndim)
               call AUXIL2(JBG,LAPL,JBH,D,X,U,V,
     1                         ndim,LAGR,Hx,Gx,n,m,l)
577            call MATSIM(A,B,C,D,LAPL,JBG,JBH,n,m,l,ndim)
               call SIMPLX(INFIX,A,B,TOL,PRM,KOUT,ERR,JH,S,
     1                         T,Y,KB,E)
c
c
      If (KOUT(1).NE.3) STOP '? Pas de sol.opt.admiss.du P.L!'
c     Y(1)=min Ezs
c
c      (calcul de la solution du probleme lineaire.)
               Do 80 k=1,4*(n+m+1)+2
                     PQR(k)=0.
                     If (KB(k).EQ.0) goto 80
                     ii=KB(k)
                     PQR(k)=S(ii)
80             continue
               Do 100 i=1,n+m+1
                     if(Dabs(PQR(i)-1.).Gt.eps2)goto 550
100            continue




c      .
c      (phase d'arret:Pt stationnaire)
c
               goto 600
550            Do 110 i=1,n
                     P(i)=PQR(i)-1.
110            continue
c
               Do 120 i=n+1,n+m
                     Q(i-n)=PQR(i)-1.
120            continue
c
               Do 130 i=n+m+1,n+m+1
                     R(i-n-m)=PQR(i)-1.
130            continue
c
               call RECLIN(X,U,V,n,m,l,ndim,P,Q,R,
     2                         itmax,alpha,eps2,Ez,Y(1),Gx,Hx,
     1                         JBG,JBH,LAGR,itest,Ichang)
c
               if((itest.EQ.0 ).OR.(itest.EQ.4)) goto 700
150    continue
c
```

```
      Write(5,7900)
7900  format(1x,'le nbre(irep) d"itera.ds P.P. est atteint',/x,
     1       ' sans avoir obtenu un Pt stationnaire ou de ',/x,
     2       ' Khun-Tucker',/x,'Desirez-vous  ausmenter ce ',
     3       ' nbre? (0=non,1=oui) ',$)
      read(5,*)it
      if(it.EQ.0) goto 700
      ideb=irep+1
      Write(5,7950)
7950  format(1x,' Nvelle valeur d"irep:',$)
      read(5,*)irep
      goto 25
c
c     (pt stationnaire)
600   itest=1
      goto 700
c
c     (Pt de K.T)
500   itest=2
700   call IMPRES(itest,ires,nbite,Ez,Y(1),Lamda,X,U,V,n,m,1,
     1            Gx,Hx)
      Write(5,8000)
8000  Format(1x,' Execution terminee.')
      call SECOND(Time2)
      Time1=Time2-Time1
      Write(ires,8100)Time1
8100  Format(/x,' Temps CPU :',f12.5)
      STOP
      END
c
c
c     ***************************************************************
      SUBROUTINE IMPRES(itest,ires,nbite,Ez,Y1,Lamda,X,U,V,n,m,1,
     1                  Gx,Hx)
c     ***************************************************************
c
c     Declarations
c     ------------




      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION Lamda,X(n),U(m),V(1),Gx(m),Hx(1)
c
c     Corps
c     -----
      Write(ires,7775)
7775  format(/x,' SOLUTION : ',/x,' ----------')
c
      if(itest.ne.4) goto 13
      Write(ires,7850)
7850  format(/x' La descente n"est pas suffisante.',/x,
     1       ' Ez ne decroit plus. Pas moyen d"obtenir un ',/x,
     2       ' pt de K.T ',/x)
      goto 250
c
13    if(itest.Ne.3) goto 70
      Write(ires,7900)
7900  format(/x,' le nbre(irep) d"iter. ds P.P est atteint',/x,
     1       ' sans obtenir un Pt stat.ou de K.T ',/x)
c
70    if(itest.NE.0) goto 200
      Write(ires,8000)
8000  format(/x,' reche.lin.interrompue car itmax est atteint',/x,
     1       'sans avoir trouve le pas Lamda.',/x)
      Write(ires,8100)Ez,Y1
8100  format(1x,' E(z)= ',D12.5,/x,'Ezs= ',D12.5)
      goto 250
```

```
c
200     if(itest.NE.1) goto 500
        Write(ires,8150)
8150    format(/x,' arret sur un Pt stationnaire! ',/x)
        s=Ez-Y1
        Write(ires,8170)s
8170    format(1x,' O(z)= ' ,D12.5)
250     Write(ires,8180)nbite
8180    format(/x,' nbre d"iterations  de l" algorithme :',I,/x)
        Do 10 i=1,n
              Write(ires,8200)i,X(i)
8200          format(1x,' X[',i2,'] =',D12.5)
10      continue
c
        Do 20 i=1,m
              Write(ires,8300)i,U(i)
8300          format(1x,' U[',i2,']= ',D12.5)
20      continue
c
        Do 30 i=1,l
              Write(ires,8400)i,V(i)
8400          format(1x,' V[',i2,']= ',D12.5)
30      continue
        Write(ires,8450)
8450    format(/x)
c
        Do 40 i=1,m
              Write(ires,8500)i,Gx(i)
8500          format(1x,' Gx[',i2,']= ',D12.5)
40      continue
c
        Do 50 i=1,l
              Write(ires,8600)i,Hx(i)
8600          format(1x,' Hx[',i2,']= ',D12.5)



50      continue
c
        goto 600
500     Write(ires,8700)
8700    format(/x,'arret sur un PT de K.T',/x)
        Write(ires,8800)Ez
8800    format(1x,'E(z)= ',D12.5)
        goto 250
600     RETURN
        END
```

```
c     ***************************************************************
      SUBROUTINE RECLIN (X,U,V,n,m,l,ndim,P,Q,R,itmax,alpha,eps2,
     1                   Ez,Ezs,Gx,Hx,JBG,JBH,LAGR,itest,ICHANG)
c     ***************************************************************
c
c     recherche lineaire le long de la direction (P,Q,R) selon la
c     regle d'Armijo.
c
c     Variables
c     ---------
c
c         Lamda : pas de la recherche lineaire.
c         Max   : valeur de E(z+lamda.s)
c         nbite : compteur du nbre d'iterations pour trouver Lamda.
c
c     Declarations
c     ------------
      implicit double precision (A-H,O-Z)
      double precision Max,Lamda,X(n),U(m),V(l),P(n),Q(m),R(l),
     1                 DF(20),LAGR(n),Gx(m),Hx(l),JBG(ndim,n),
     2                 JBH(ndim,n)
c
c     Corps
c     -----
      Lamda=1.
      Y=alpha*(Ez-Ezs)
      Do 20 i=1,n
            X(i)=X(i)+P(i)
20    continue
c
      Do 30 i=1,m
            U(i)=U(i)+Q(i)
30    continue
c
      Do 40 i=1,l
            V(i)=V(i)+R(i)
40    continue
c
      ideb=1
45    Do 100 i=ideb,itmax
            nbite=i
            call CINEG (X,Gx,n,m)
            call CEGAL (X,Hx,n,l)
            call GRADF (X,DF,n)
            call JACOG (X,JBG,ndim,n)
            call JACOH (X,JBH,ndim,n)
            call LAGRAN(DF,JBG,JBH,U,V,n,m,l,ndim,LAGR)
            call FCTOBJ(LAGR,Hx,Gx,X,U,V,n,m,l,Max,ichang)
c
c
c
c
c
c
            If ((Ez-Max).GE.LAMDA*LAMDA*Y) Goto 200
            Lamda=Lamda*0.5
            Do 50 j=1,n
                  X(j)=X(j)-Lamda*P(j)
50          continue
c
            Do 60 j=1,m
                  U(j)=U(j)-Lamda*Q(j)
60          continue
c
            Do 70 j=1,l
                  V(j)=V(j)-Lamda*R(j)
70          continue
100   continue
```

```
c
      Write(5,8000)
8000  Format(1x,' Le nbre max d"iterations etant depasse ds ',/x
     1       'la recherche lineaire,desirez-vous augmenter ',
     2       'ce nbre ?',/6x,'(0=non,1=oui) ',$)
c
      ideb=itmax+1
      read(5,*)it
      if(it.EQ.0) goto 250
      Write (5,8100)
8100  Format(1x,' Nvelle valeur de itmax : ',$)
      read(5,*)itmax
      goto 45
200   if((Ez-Max).gt.eps2) goto 201
      itest=4
      return
201   Ez=Max
      Write(5,7200)Ez
7200  format(1x,'Ez= ',D13.6)
      itest=3
      goto 500
250   itest=0
c
500   RETURN
      END
c
c
c     ***************************************************
      SUBROUTINE LAGRAN (A,B,C,U,V,n,m,l,ndim,LAGR)
c     ***************************************************
c
c     calcul du Lagransien
c
c     Variables
c     ---------
c        A <--- DF
c        B <--- JBG
c        C <--- JBH
c
c     Declarations
c     ------------
      implicit double precision (A-H,O-Z)
      double precision A(n),U(m),V(l),LAGR(n),S(20),B(ndim,n),
     1                 C(ndim,n)
c
c     Corps
c     -----




      call PS1 (B,U,m,n,l,ndim,LAGR)
      call PS1 (C,V,l,n,l,ndim,S)
c
      Do 40 i=1,n
            LAGR(i)=LAGR(i)+A(i)+S(i)
40    continue
      RETURN
      END
c
c
c     ***************************************************
      SUBROUTINE LAPLAC(A,B,C,U,V,n,m,l,ndim,LAPL)
c     ***************************************************
c
c     calcul du Laplacien.
c
```

```
c       Variables
c       ---------
c            A <----- LAPLF
c            B <----- LAPLG
c            C <----- LAPLH
c
c       Declarations
c       ------------
        IMPLICIT DOUBLE PRECISION(A-H,O-Z)
        DOUBLE PRECISION A(ndim,n),B(ndim,n),C(ndim,n),U(m),V(1),
        1                LAPL(ndim,n)
c
c       Corps
c       -----
        Do 40 i=1,n
              Do 50 J=1,n
                    LAPL(i,J)=0.
                    Do 60 k=0,m-1
                          LAPL(i,J)=B(i+n*k,J)*U(k+1)+LAPL(i,J)
60                  continue
                    Do 70 k=0,l-1
                          LAPL(i,J)=LAPL(i,J)+C(i+n*k,J)*V(k+1)
70                  continue
              LAPL(i,J)=LAPL(i,J)+A(i,J)
50            continue
40      continue
        RETURN
        END

c
c
c       ***************************************************
        SUBROUTINE AUXIL(JBG,LAPL,JBH,D,X,U,V,ndim,LAGR,Hx,
        1                Gx,n,m,l)
c       ***************************************************
c
c       sert a calculer une des lignes de la matrice du
c       Simplex associe au Pb (3).
c
c       Declarations
c       ------------
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DOUBLE PRECISION JBG(ndim,n),LAPL(ndim,n),JBH(ndim,n),
        1                D(ndim),X(n),V(1),U(m),LAGR(n),Hx(1),
        2                Gx(m),S1(20),S2(20),S3(20)
c




c       Corps
c       -----
        call PS1(JBG,U,m,n,1,ndim,S1)
        call PS1(LAPL,X,n,n,0,ndim,S2)
        call PS1(JBH,V,1,n,1,ndim,S3)
c
        Do  20 i=1,n
              D(i)=-S1(i)-S3(i)+LAGR(i)+S2(i)
20      continue
        call PS1(JBG,X,m,n,0,ndim,S1)
c
        Do 80 i=1,m
             D(i+n)=-Gx(i)+S1(i)
80      continue
        call PS1(JBH,X,1,n,0,ndim,S1)
c
        Do 100 i=1,l
              D(i+n+m)=-Hx(i)+S1(i)
```

```
100     continue
        D(i+n+m+1)=-1.
c
        RETURN
        END
c
c
c       ***************************************************
        SUBROUTINE TERIND(LAGR,LAPL,JBG,JBH,Gx,Hx,X,U,V,
        1                 C,n,m,l,ndim)
c       ***************************************************
c
c       calcule une partie du terme independant du Simplex
c
c       Declarations
c       ------------
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DOUBLE PRECISION LAGR(n),Gx(m),Hx(1),A(20),X(n),U(m),V(1),
        1                C(ndim),JBG(ndim,n),JBH(ndim,n),
        2                LAPL(ndim,n)
c
c       Corps
c       -----
        S1=0.
        Do 10 i=1,n
              C(i)=0.
              Do 15 k=1,m
                    C(i)=C(i)+JBG(k,i)
15            continue
c
              Do 17 k=1,l
                    C(i)=C(i)+JBH(k,i)
17            continue
c
              Do 19 k=1,n
                    C(i)=C(i)+LAPL(k,i)
19            continue
c
              C(i)=C(i)-LAGR(i)
              S1=S1+LAGR(i)
10      continue
c
        call FS2(C,X,n,S2)




        Do 20 i=n+1,n+m
              C(i)=0.
              Do 25 k=1,n
                    C(i)=C(i)+JBG(i-n,k)
25            continue
              C(i)=C(i)-Gx(i-n)
20      continue
c
        Do 30 i=m+n+1,n+m+l
              C(i)=0.
              Do 35 k=1,n
                    C(i)=C(i)+JBH(i-n-m,k)
35            continue
              C(i)=C(i)-Hx(i-n-m)
30      continue
c
        Do 40 i=n+m+l+1,n+m+m+l
              C(i)=U(i-n-m-l)-1.
40      continue
c
```

```
      Do 50 j=1,m
            A(j)=-C(j+n)
            S1=S1-Gx(j)
50    continue
      call PS2(A,U,m,S3)
      Do 60 j=1,l
            A(j)=-C(j+n+m)
            S1=S1-Hx(j)
60    continue
      call PS2(A,V,l,S4)
      C(n+2*m+l+1)=S1+S2+S3+S4
      RETURN
      END
```

```
c     ************************************************************
      SUBROUTINE DONNEE (ires,n,m,l,ichang,X,U,V,eps1,eps2,
     1                   alpha)
c     ************************************************************
c
c     ecriture des donnees dans le fichier resultat
c
c     Declarations
c     ------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION X(n),U(m),V(l)
c
c     Corps
c     -----
      Write(ires,9000)
9000  Format(1x,'DONNEES DE DEPART:',/x,18('-'),/)
      Write(ires,9050)n,m,l,ichang
9050  Format(1x,'n= ',i2,x,'m= ',i2,x,'l= ',i2,/x,
     1       'ichang= ',i1,/x)
      Do 10 i=1,n
            Write(ires,9100)i,X(i)
9100        Format(1x,'X[',i2,']= ',D12.5)
10    continue
c
```

```
      Do 20 i=1,m
            Write(ires,9150)i,U(i)
9150        Format(1x,'U[',i2,']= ',D12.5)
20    continue
c
      Do 30 i=1,l
            Write(ires,9200)i,V(i)
9200        Format(1x,'V[',i2,']= ',D12.5)
30    continue
```

```
c
      Write(ires,9250)eps1,eps2
9250  Format(/x,'Eps1= ',D12.5,3x,'Eps2= ',D12.5)
      Write(ires,9300)alpha
9300  Format(/x,'alpha= ',D12.5)
c
      RETURN
      END
c
c
c     **********************
      SUBROUTINE PS2 (W,V,n,S)
c     **********************
c
c     produit scalaire de 2 vect. de dim n
c
c     Declarations
c     ------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION W(n),V(n)
c
c     Corps
c     -----
      S=0.
      Do 20 i=1,n
            S= S+W(i)*V(i)
20    continue
      RETURN
      END
c
c
c     ************************************
      SUBROUTINE PS1 (W,V,IL,IC,ierr,ndim,S)
c     ************************************
c
c     Declarations
c     ------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION V(1),S(1),W(ndim,IC)
c
c     produit scalaire de W d"ordre(IL,IC) par le vect.V
c     ierr=1 si W transposee
c         =0 sinon
c     S:resultat du prod.scal.
c
c     Corps.
c     ------
      If (ierr.EQ.1) goto 40
      DO 20 i=1,IL
            S(i)= 0.
            DO 30 J=1,IC
                  S(i)= S(i)+ W(i,J)*V(J)




30          Continue
20    Continue
      goto 70
40    DO 50 i=1,IC
            S(i)= 0.
            DO 60 J=1,IL
                  S(i)= S(i)+ W(J,i)*V(J)
60          Continue
50    Continue
70    RETURN
      END
```

```
c
c
c       ***********************************************************
        SUBROUTINE FCTOBJ (LAGR,Hx,Gx,X,U,V,n,m,l,MAX,ichang)
c       ***********************************************************
c
c
c       calcul de E(z)
c
c       Declarations
c       ------------
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DOUBLE PRECISION MAX,LAGR(n)
        DIMENSION Hx(l),Gx(m),X(n),U(m),V(l)
c
c       Corps
c       -----
        MAX=0.
        Do 10 i=1,n
              if ( Dabs(LAGR(i)).GT.MAX ) MAX=Dabs(LAGR(i))
10      Continue
c
        Do 20 i=1,l
              if ( Dabs(Hx(i)).GT.MAX ) MAX=Dabs(Hx(i))
20      Continue
c
        Do 30 i=1,m
              if ((-U(i)).GT.MAX) MAX=-U(i)
              if (Gx(i).GT.MAX) MAX=Gx(i)
30      Continue
c
        S1=0.
        S2=0.
        if (m.Ne.0) call PS2(Gx,U,m,S1)
        if(ichang.eq.1) goto 40
        if (l.Ne.0) call PS2(Hx,V,l,S2)
        Call PS2(LAGR,X,n,S3)
        if ((-S1-S2+S3).GT.MAX) MAX=-S1-S2+S3
        RETURN
40      if ((-S1).gT.MAX) MAX=-S1
        RETURN
        END


@
```

```
c       ***********************************************************
        SUBROUTINE AUXIL2(JBG,LAPL,JBH,D,X,U,V,ndim,LAGR,Hx,
        1                 Gx,n,m,l)
c       ***********************************************************



c
c       calcule une partie de ligne de la matrice A du Simplex
c       associe au pb (3)
c
c       Declarations
c       ------------
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DOUBLE PRECISION JBG(ndim,n),LAPL(ndim,n),JBH(ndim,n),
        1                D(ndim),X(n),V(1),U(m),LAGR(n),Hx(1),
        2                Gx(m),S1(20),S2(20),S3(20)
c
c       Corps
c       -----
        call PS1(JBG,U,m,n,1,ndim,S1)
c
        Do  20 i=1,n
              D(i)=-S1(i)
20      continue
c
        Do 80 i=1,m
             D(i+n)=-Gx(i)
80      continue
c
        Do 100 i=1,l
              D(i+n+m)=0.
100     continue
        D(l+n+m+1)=-1.
c
        RETURN
        END
c
c
c       ***********************************************************
        SUBROUTINE TERIN2(LAGR,LAPL,JBG,JBH,Gx,Hx,X,U,V,
        1                  C,n,m,l,ndim)
c       ***********************************************************
c
c       calcule une partie du terme independant du Simplex
c
c       Declarations
c       ------------
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DOUBLE PRECISION LAGR(n),Gx(m),Hx(1),A(20),X(n),U(m),V(1),
        1                C(ndim),JBG(ndim,n),JBH(ndim,n),
        2                LAPL(ndim,n)
c
c       Corps
c       -----
        S1=0.
        Do 10 i=1,n
              C(i)=0.
              Do 15 k=1,m
```

```
                C(i)=C(i)+JBG(k,i)
15          continue
c
            Do 17 k=1,l
                C(i)=C(i)+JBH(k,i)
17          continue
c
            Do 19 k=1,n
                C(i)=C(i)+LAPL(k,i)




19          continue
c
            C(i)=C(i)-LAGR(i)
            S1=S1+LAGR(i)
10      continue
c
        call PS2(C,X,n,S2)
        Do  20 i=n+1,n+m
            C(i)=0.
            Do 25 k=1,n
                C(i)=C(i)+JBG(i-n,k)
25          continue
            C(i)=C(i)-Gx(i-n)
20      continue
c
        Do 30 i=m+n+1,n+m+l
            C(i)=0.
            Do 35 k=1,n
                C(i)=C(i)+JBH(i-n-m,k)
35          continue
            C(i)=C(i)-Hx(i-n-m)
30      continue
c
        Do 40 i=n+m+l+1,n+m+m+l
            C(i)=U(i-n-m-l)-1.
40      continue
c
        Do 50 J=1,m
            A(J)=-C(J+n)
50      continue
        call PS2(A,U,m,S3)
c
        S2=0.
        Do 60 i=1,m
            S2=S2-Gx(i)
60      continue
c
        C(n+2*m+1+1)=S3 + S2
        RETURN
        END

@
```

itmax : nombre max. d'iterations permis dans la recherche
lineaire pour determiner le pas d'Armijo
nbite : compte le nombre d'iterations mis pour trouver le
pas d'Armijo
Eps1 : x = 0 a Eps1.pres
Eps2 : x-y = 0 a Eps2.pres
itest : variable test initialisee a 3
a la fin de l'execution
si itest = 3 --> le nbre d'iter. max. de l'algo. est
atteint sans avoir obtenu de Pt.
stationnaire ou de K.K.T.
itest = 2 --> arret sur un Pt. de K.K.T.
itest = 1 --> arret sur un Pt. stationnaire
itest = 0 --> recherche lineaire interrompue
itest = 4 --> la decroissance de E(z) n'est pas
suffisante
X : vecteur de dimension n contenant les n variables du pb (1)
U : vecteur des multiplicateurs de Lagrange associes aux
m contraintes d'inegalites
V : vecteur des multiplicateurs de Lagrange associes aux

l contraintes d'egalites

not
notation : (X,U,V) = z
Gx : vecteur des m contraintes d'inegalites calculees en X
Hx : vecteur des l contraintes d'egalites calculees en X
DF : GradF(x) ou F fonction objectif du pb (1)
JBG : matrice Jacobienne de Gx en X
JBH : matrice Jacobienne de Hx en X
LAGR : Lagrangien au pt. z du pb (1)
DF + JBG'.U + JBH'.V
LAPLF : matrice hessienne de F en X
LAPLG : matrice d'ordre (n,m,n) contenant le Hessien associe
a chaque contrainte d'inegalite
LAPLH : matrice d'ordre (n,l,n) contenant le Hessien associe
a chaque contrainte d'egalite
LAPL : Laplacien correspondant

not
notation : (P,Q,R) = S --> direction de descente de E(z)
au pt z
P : vecteur de dim n correspondant a X
Q : vecteur de dim m correspondant a U
R : vecteur de dim l correspondant a V
PQR : vecteur de dim (n+m+l) solution du pb (3)
ta PQR(i) = P(i) + 1 (i=1,n)
PQR(i) = Q(i) + 1 (i=n+1,n+m)
PQR(i) = R(i) + 1 (i=n+m+1,n+m+l)
A : matrice du simplexe associe au pb (3)
B : vecteur independant du meme pb
C : vecteur reprenant certains composantes du vect. B
D : vecteur contenant une partie de ligne de A

BIBLIOGRAPHIE.

(1) M.R. HESTENES, Calculus of Variations and Optimal Control Theory, John Wiley and Sons, Inc., New York, 1966.

(2) W. HOCK - K. SCHITTKOWSKI, Test examples for Nonlinear Programming Codes, by Springer-Verlag, Berlin Heidelberg, New York, 1981.

(3) MORDECAI AVRIEL, Non LINEAR Programming : Analysis and Methods, Prentice Hall, INC., Englewood Cliffs, New Jersey, 1976.

(4) R.T. ROCKAFELLAR, Convex Analysis Princeton University Princeton, New Jersey, 1970.

(5) J. KREUSER, Superlinearly Globally Convergent Algorithms for Nonlinear programming via sequential linear programs, a thesis submitted to the Graduate School of the University of Wisconsin - Madison, 1979.