



HAL
open science

Contribution à l'étude des algorithmes d'optimisation en analyse des données

Mohamed Ali Boughazi

► **To cite this version:**

Mohamed Ali Boughazi. Contribution à l'étude des algorithmes d'optimisation en analyse des données. Modélisation et simulation. Université Joseph-Fourier - Grenoble I, 1987. Français. NNT: . tel-00323967

HAL Id: tel-00323967

<https://theses.hal.science/tel-00323967v1>

Submitted on 23 Sep 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THESE

présentée à

**L'UNIVERSITE SCIENTIFIQUE, TECHNOLOGIQUE et MEDICALE de
GRENOBLE .**

*pour obtenir le grade de
DOCTEUR de 3^{ème} CYCLE
Mathématiques Appliquées*

par

Mohamed Ali BOUGHAZI

sujet

**CONTRIBUTION A L'ETUDE DES ALGORITHMES
D'OPTIMISATION EN ANALYSE DES DONNEES.**

Thèse soutenue le 30 avril 1987 devant la commission d'examen .

Monsieur P.J.LAURENT

Président

**Messieurs H.ATTOUCH
G.DROUET D'AUBIGNY
B.LEMAIRE
PHAM DINH TAO
F.ROBERT**

Examineurs



UNIVERSITE SCIENTIFIQUE TECHNOLOGIQUE ET MEDICALE DE GRENOBLE

Président de l'Université :
M. TANCHÉ

Année Universitaire 1986 - 1987

MEMBRES DU CORPS ENSEIGNANT DE SCIENCES ET DE GEOGRAPHIE

PROFESSEURS DE 1ère Classe

ARNAUD Paul	Chimie Organique
ARVIEU ROBERT	Physique Nucléaire I.S.N.
AUBERT Guy	Physique C.N.R.S
AURIAULT Jean-Louis	Mécanique
AYANT Yves	Physique Approfondie
BARBIER Marie-Jeanne	Electrochimie
BARBIER Jean-Claude	Physique Expérimentale CNRS
BARJON Robert	Physique Nucléaire ISN
BARNOUD Fernand	Bochimie Macromoléculaire Végétale
BARRA Jean-René	Statistiques-Mathématiques Appliquées
BELORISKY Elie	Physique C.E.N.G- D.R.F.
BENZAKEN Claude	Mathématiques Pures
BERNARD Alain	Mathématiques Pures
BERTRANDIAS Françoise	Mathématiques Pures
BERTRANDIAS Jean-Paul	Mathématiques Pures
BILLET Jean	Géographie
BOELHER Jean-Paul	Mécanique
BONNIER Jane Marie	Chimie Générale
BOUCHEZ Robert	Physique Nucléaire ISN
BRAVARD Yves	Géographie
CARLIER Georges	Biologie Végétale
CAUQUIS Georges	Chimie Organique
CHIBON Pierre	Biologie Animale
COHEN ADDAD Jean-Pierre	Physique
COLIN DE VERDIERE Yves	Mathématiques Pures
CYROT Michel	Physique du Solide
DEBELMAS Jacques	Géologie Générale
DEGRANGE Charles	Zoologie
DELOBEL Claude	Mathématiques Appliquées
DEPORTES Charles	Chimie Minérale
DESRE Pierre	Electrochimie
DOLIQUE Jean-Michel	Physique des Plasmas
DOUCE Rolland	Physiologie Végétale
DUCROS Pierre	Cristallographie
FONTAINE Jean-Marc	Mathématiques Pures
GAGNAIRE Didier	Chimie Physique
GERMAIN Jean-Pierre	Mécanique,
GIRAUD Pierre	Géologie
HICTER Pierre	Chimie
IDELMAN Simon	Physiologie Animale
JANIN Bernard	Géographie
JOLY Jean-René	Mathématiques Pures
KAHANE André, détaché	Physique
KAHANE Josette	Physique
KRAKOWIAK Sacha	Mathématiques Appliquées
KUPKA Yvon	Mathématiques Pures
LAJZEROWICZ Jeanine	Physique
LAJZEROWICZ Joseph	Physique
LAURENT Pierre-Jean	Mathématiques Appliquées
DE LEIRIS Joel	Biologie

LLIBOUTRY Louis
 LOISEAUX Jean-Marie
 MACHE Régis
 MAYNARD Roger
 MICHEL Robert
 OMONT Alain
 OZENDA Paul
 PAYAN Jean-Jacques
 PEBAY-PEYROULA Jean-Claude
 PERRIAUX Jacques
 PERRIER Guy
 PIERRARD Jean-Marie
 PIERRE Jean-Louis
 RASSAT André
 RENARD Michel
 RINAUDO Marguerite
 ROSSI André
 SAKAROVITCH Michel
 SAXOD Raimard
 SENDEL Philippe
 SERGERAERT Francis
 SOUCHIER Bernard
 SOUTIF Michel
 STUTZ Pierre
 VALENTIN Jacques
 VAN CUTSEM Bernard
 VIALON Pierre

Géophysique
 Sciences Nucléaires I.S.N.
 Physiologie Végétale
 Physique du Solide
 Minéralogie et Pétrographie (Géologie)
 Astrophysique
 Botanique (Biologie Végétale)
 Mathématiques Pures
 Physique
 Géologie
 Géophysique
 Mécanique
 Chimie Organique
 Chimie Systématique
 Thermodynamique
 Chimie CERMAV
 Biologie
 Mathématiques Appliquées
 Biologie Animale
 Biologie Animale
 Mathématiques Pures
 Biologie
 Physique
 Mécanique
 Physique Nucléaire I.S.N.
 Mathématiques Appliquées
 Géologie

PROFESSEURS de 2^{ème} Classe

ADIBA Michel
 ANTOINE Pierre
 ARMAND Gilbert
 BARET Paul
 BECKER Pierre
 BEGUIN Claude
 BLANCHI J.Pierre
 BOITET Christian
 BORNAREL Jean
 BRUANDET J.François
 BRUN Gilbert
 CASTAING Bernard
 CERFF Rudiger
 CHARDON Michel
 CHIARAMELLA Yves
 COURT Jean
 DEMAILLY Jean-Pierre
 DENEUVILLE Alain
 DEPASSEL Roger
 DERRIEN Jacques
 DUFREYNOY Alain
 GASPARD François
 GAUTRON René
 GENIES Eugène
 GIDON Maurice
 GIGNOUX Claude
 GILLARD Roland
 GIORNI Alain
 GUIGO Maryse
 GUMUCHAIN Hervé
 GUITTON Jacques
 HACQUES Gérard

Mathématiques Pures
 Géologie
 Géographie
 Chimie
 Physique
 Chimie Organique
 STAPS
 Mathématiques Appliquées
 Physique
 Physique
 Biologie
 Physique
 Biologie
 Géographie
 Mathématiques Appliquées
 Chimie
 Mathématiques Pures
 Physique
 Mécanique des Fluides
 Physique
 Mathématiques Pures
 Physique
 Chimie
 Chimie
 Géologie
 Sciences Nucléaires
 Mathématiques Pures
 Sciences Nucléaires
 Géographie
 Géographie
 Chimie
 Mathématiques Appliquées

HERBIN Jacky
 HERAULT Jeanny
 JARDON Pierre
 JOSELEAU Jean-Paul
 KERCKHOVE Claude
 LEBRETON Alain
 LONGEQUEUE Nicole
 LUCAS Robert
 LUNA Domingo
 MANDARON Paul
 MARTINEZ Francis
 MASCLE Georges
 NEMOZ Alain
 OUDET Bruno
 PELMONT Jean
 PERRIN Claude
 PFISTER Jean-Claude
 PIBOULE Michel
 RAYNAUD Hervé
 RIEDIMANN Christine
 ROBERT Gilles
 ROBERT Jean-Bernard
 SARROT-REYNAULD Jean
 SAYETAT Françoise
 SERVE Denis
 STOECKEL Frédéric
 SOUTIF Jeanne
 SCHOLL Pierre-Claude
 SUBRA Robert
 VALLADE Marcel
 VIDAL Michel
 VIVIAN Robert
 VOTTERO Philippe

Géographie
 Physique
 Chimie
 Biochimie
 Géologie
 Mathématiques Appliquées
 Sciences Nucléaires I.S.N.
 Physique
 Mathématiques Pures
 Biologie
 Mathématiques Appliquées
 Géologie
 Thermodynamique CNRS - CRTBT
 Mathématiques Appliquées
 Biochimie
 Sciences Nucléaires I.S.N.
 Physique du Solide
 Géologie
 Mathématiques Appliquées
 Mathématiques Pures
 Mathématiques Pures
 Chimie Physique
 Géologie
 Physique
 Chimie
 Physique
 Physique
 Mathématiques Appliquées
 Chimie
 Physique
 Chimie Organique
 Géographie
 Chimie

MEMBRES DU CORPS ENSEIGNANT DEL' IUT 1

PROFESSEURS de 1^{ère} Classe

BUISSON Roger
 DODU Jacques
 NEGRE Robert

Physique IUT 1
 Mécanique Appliquée IUT 1
 Génie Civil IUT 1

PROFESSEURS de 2^{ème} classe

BOUTHINON Michel
 CHAMBON René
 CHEHIKIAN Alain
 CHENAVAS Jean
 CHOUTEAU Gérard
 CONTE René
 GOSSE Jean-Pierre
 GROS Yves
 KUIHN Gérard, (Détaché)
 MAZUER Jean
 MICHOUILLER Jean
 MONLLOR Christian
 NOUGARÉT Marcel
 PEFFEN René
 PERARD Jacques
 PERRAUD Robert
 TERRIEZ Jean-Michel
 TOUZAIN Philippe
 VINCENDON Marc

EEA. IUT 1
 Génie Mécanique IUT 1
 EEA. IUT 1
 Physique IUT 1
 Physique IUT 1
 Physique IUT 1
 EEA.IUT 1
 Physique IUT 1
 Physique IUT 1
 Physique IUT 1
 EEA.IUT 1
 Automatique IUT 1
 Métallurgie IUT 1
 EEA. IUT 1
 Chimie IUT 1
 Génie Mécanique IUT 1
 Chimie IUT 1
 Chimie IUT 1

MEMBRES DU CORPS ENSEIGNANT DE MEDECINE

PROFESSEURS CLASSE EXCEPTIONNELLE ET 1ère CLASSE

AMBLARD Pierre	Dermatologie	C.H.R.G.
AMBROISE-THOMAS Pierre	Parasitologie	C.H.R.G.
BEAUDOING André	Pédiatrie-Puericulture	C.H.R.G.
BEZEZ Henri	Orthopédie-Traumatologie	Hopital SUD
BONNET Jean-Louis	Ophthalmologie	C.H.R.G.
BOUCHET Yves	Anatomie	Faculté La Merci
	Chirurgie Générale et Digestive	C.H.R.G.
BUTEL Jean	Orthopédie-Traumatologie	C.H.R.G.
CHAMPETIER Jean	Anatomie-Topographique et Appliquée	C.H.R.G.
	O.R.L.	C.H.R.G.
CHARACHON Robert	Anatomie-Pathologique	C.H.R.G.
COUDERC Pierre	Pneumophtisiologique	C.H.R.G.
DELORMAS Pierre	Cardiologie	C.H.R.G.
DENIS Bernard	Pharmacologie	Faculté La Merci
GAVEND Michel	Hématologie	C.H.R.G.
HOLLARD Daniel	Chirurgie Thoracique et Cardiovasculaire	C.H.R.G.
LATREILLE René	Bactériologie-Virologie	C.H.R.G.
	Gynécologie et Obstétrique	C.H.R.G.
LE NOC Pierre	Médecine du Travail	C.H.R.G.
MALINAS Yves	Clinique Médicale et Maladies Infectieuses	C.H.R.G.
MALLION Jean-Michel	Histologie	Faculté La Merci
MICOUD Max	Pneumologie	C.H.R.G.
	Neurologie	C.H.R.G.
MOURIQUAND Claude	Hépto-Gastro-Entérologie	C.H.R.G.
PARAMELLE Bernard	Neurochirurgie	C.H.R.G.
PERRET Jean	Clinique Chirurgicale	C.H.R.G.
RACHAIL Michel	Anesthésiologie	C.H.R.G.
DE ROUGEMONT Jacques	Physiologie	Faculté La Merci
SARRAZIN Roger	Biophysique	Faculté La Merci
STIEGLITZ Paul	Biochimie	Faculté La Merci
TANCHE Maurice		
VERAIN André		
VIGNAIS Pierre		

PROFESSEURS 2ème CLASSE

BACHELOT Yvan	Endocrinologie	C.H.R.G.
BARGE Michel	Neurochirurgie	C.H.R.G.
BENABID Alim Louis	Biophysique	Faculté La Merci
BENSA Jean-Claude	Immunologie	Hopital Sud
BERNARD Pierre	Gynécologie-Obstétrique	C.H.R.G.
BESSARD Germain	Pharmacologie	ABIDJAN
BOLLA Michel	Radiothérapie	C.H.R.G.
BOST Michel	Pédiatrie	C.H.R.G.
BOUCHARLAT Jacques	Psychiatrie Adultes	Hopital Sud
BRAMBILLA Christian	Pneumologie	C.H.R.G.
CHAMBAZ Edmond	Biochimie	C.H.R.G.
CHIROSEL Jean-Paul	Anatomie-Neurochirurgie	C.H.R.G.
COLOMB Maurice	Immunologie	Hopital Sud
COMET Michel	Biophysique	Faculté La Merci
CONTAMIN Charles	Chirurgie Thoracique et Cardiovasculaire	C.H.R.G.
	Néphrologie	C.H.R.G.
CORDONNIER Daniel	Radiologie	C.H.R.G.
COULOMB Max	Radiologie	C.H.R.G.
CROUZET Guy	Radiologie	C.H.R.G.
DEBRU Jean-Luc	Médecine Interne et Toxicologie	C.H.R.G.
DEMONGEOT Jacques	Biostatistiques et Informatique Médicale	Faculté La Merci

DUPRE Alain	Chirurgie Générale	C.H.R.G.
DYON Jean-François	Chirurgie Infantile	C.H.R.G.
ETERRADOSSI Jacqueline	Physiologie	Faculté La Merci
FAURE Claude	Anatomie et Organogénèse	C.H.R.G.
FAURE Gilbert	Urologie	C.H.R.G.
FOURNET Jacques	Hépto-Gastro-Entérologie	C.H.R.G.
FRANCO Alain	Médecine Interne	C.H.R.G.
GIRARDET Pierre	Anesthésiologie	C.H.R.G.
GUIDICELLI Henri	Chirurgie Générale et Vasculaire	C.H.R.G.
GUIGNIER Michel	Thérapeutique et Réanimation Médicale	C.H.R.G.
HADJIAN Arthur	Biochimie	Faculté La Merci
HALIMI Serge	Endocrinologie et Maladies Métaboliques	C.H.R.G.
HOSTEIN Jean	Hépto-Gastro-Entérologie	C.H.R.G.
HUGONOT Robert	Médecine Interne	C.H.R.G.
JALBERT Pierre	Histologie-Cytogénétique	C.H.R.G.
JUNIEN-LAVILLAURROY Claude	O.R.L.	C.H.R.G.
KOLODIE Lucien	Hématologie Biologique	C.H.R.G.
LETOUBLON Christian	Chirurgie Générale	C.H.R.G.
MACHIECOURT Jacques	Cardiologie et Maladies Vasculaires	C.H.R.G.
MAGNIN Robert	Hygiène	C.H.R.G.
MASSOT Christian	Médecine Interne	C.H.R.G.
MOUILLON Michel	Ophthalmologie	C.H.R.G.
PELLAT Jacques	Neurologie	C.H.R.G.
PHILIP Xavier	Rhumatologie	C.H.R.G.
RACINET Claude	Gynécologie	C.H.R.G.
RAMBAUD Pierre	Pédiatrie	C.H.R.G.
RAPHAEL Bernard	Stomatologie	C.H.R.G.
SCHAERER René	Cancérologie	C.H.R.G.
SEIGNEURIN Jean-Marie	Bactériologie-Virologie	Faculté La Merci
SELE Bernard	Cytogénétique	Faculté La Merci
SOTTO Jean-Jacques	Hématologie	C.H.R.G.
STOEBNER Pierre	Anatomie Pathologique	C.H.R.G.
VROUSOS Constantin	Radiothérapie	C.H.R.G.

MEMBRES DU CORPS ENSEIGNANT PHARMACIE

AGNIUS-DELORD Claudine	Physique	Faculté La Tronche
ALARY Josette	Chimie Analytique	Faculté La Tronche
BERIEL Hélène	Physiologie et Pharmacologie	Faculté La Tronche
BOUCHERLE André	Chimie et Toxicologie	Faculté Meylan
CUSSAC Max	Chimie Thérapeutique	Faculté La Tronche
DEMENGE Pierre	Pharmacodynamie	Faculté La Tronche
JEANNIN Charles	Pharmacie Galénique	Faculté Meylan
LATURAZE Jean	Biochimie	Faculté La Tronche
LUU DUC Cuong	Chimie Générale	Faculté La Tronche
MARIOTTE Anne-Marie	Pharmacognosie	Faculté La Tronche
MARZIN Daniel	Toxicologie	Faculté Meylan
RENAUDET Jacqueline	Bactériologie	Faculté La Tronche
ROCHAT Jacques	Hygiène et Hydrologie	Faculté La Tronche
SEIGLE-MURANDI Françoise	Botanique et Cryptogamie	Faculté Meylan
VERAIN Alice	Pharmacie Galénique	Faculté Meylan

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

الهدى

إلى الأم الحنون التي أمدتني من نور قلبها يوم أرضعتني من حليبها .
إلى الأب الكريم الذي طعم عقلي بالعلم والإيمان وغذاني بالحلال .
" رَبِّ ارْحَمْهُمَا كَمَا رَبَّيْتَانِي صَغِيرًا "

إلى الذين كان وجودهم سكنا وراحة لي : أم أيمن وإيمان وأيمن .
إلى اللذين كانا لي مربيين في صغري ونموذجا في حياتي ومونا لي
في كل أمور : أخي مختار وأخي عبد الناصر .
إلى التي كانت بجانب طيلة حياتي الجامعية أخي أم جهاد .
إلى اللذين لم يبخلا علي بنصائحهما وإرشاداتهما أبو مصعب
و"لموشى فؤاد".
إلى كل اخوتي واخواتي الذين كان حنانهم سرائي نجاحي .
إلى أسرة خلفي وويشاوي وأخص بالذكر عمي أحسن والحاج مبروك "

أبو بكر



A mon père.
A ma mère.
A ma femme.
A mes enfants, Imane et Ayman.
A mes frères et sœurs.



Cette thèse a été préparée au laboratoire TIM3 de l'Institut d'Informatique et de Mathématiques Appliquées de Grenoble.

Je tiens à exprimer toute ma reconnaissance à Monsieur PHAM DINH TAO qui a guidé mes premiers pas dans la recherche, pour son soutien, ses conseils et sa bienveillance durant l'élaboration de cette thèse.

Je suis très sensible à l'honneur que me fait Monsieur P.J.LAURENT en acceptant de présider ce jury. Qu'il veuille bien agréer l'expression de mes sincères et respectueuses considérations.

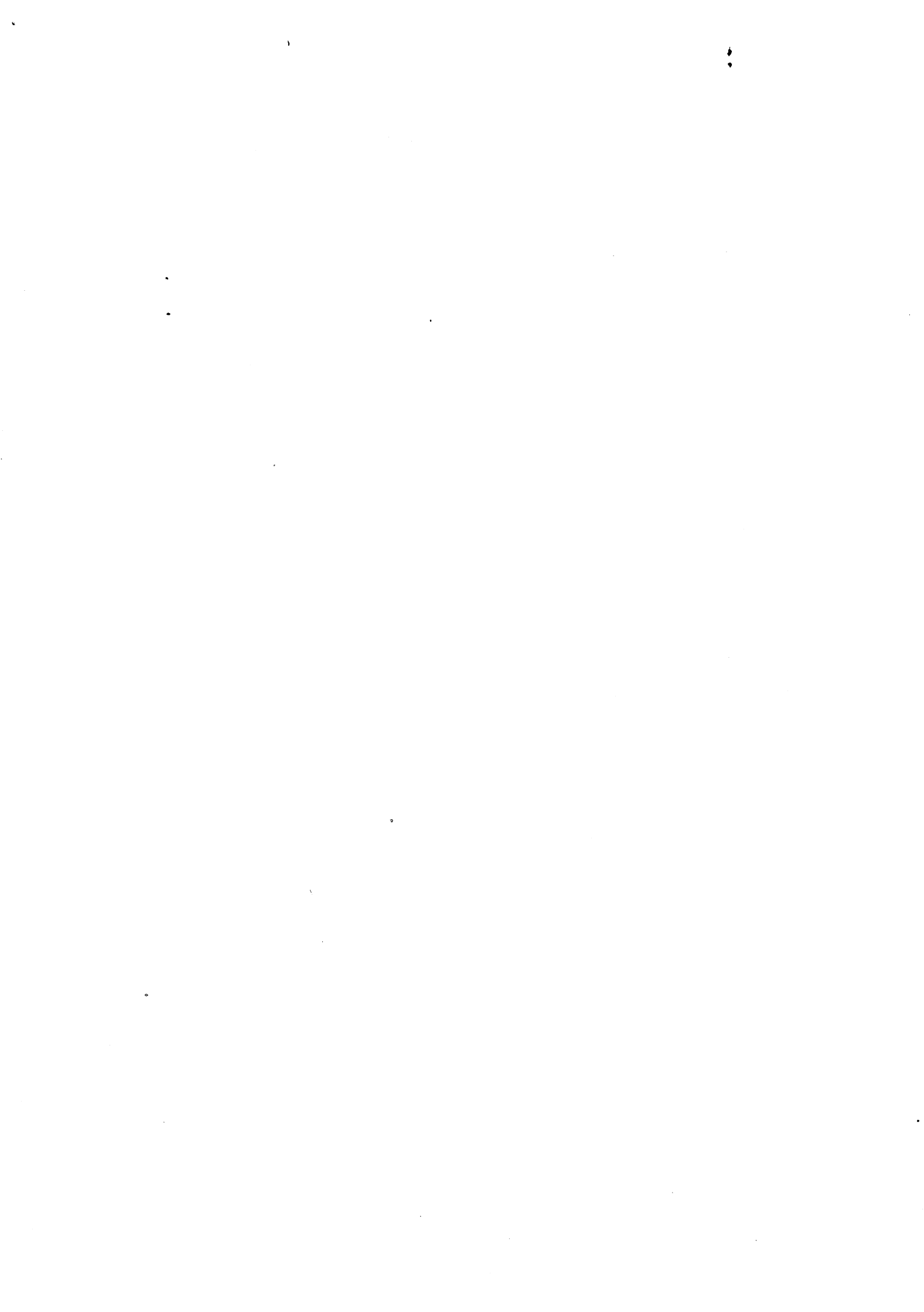
Je témoigne tout ma gratitude à Monsieur G.DROUOT D'AUBIGNY, Directeur du département de statistique, pour les discussions fructueuses que nous avons eues dans le domaine de l'Analyse des Données, l'intérêt qu'il a manifesté à ce travail et l'aide précieuse durant la rédaction de cette thèse.

Messieurs H.ATTOUCH, Professeur à l'Université de Perpignan et B.LEMAIRE, Professeur à l'Université de Montpellier ont bien voulu juger ce travail et venir à Grenoble faire partie du jury. Qu'ils veuillent bien accepter mes sincères et respectueux remerciements.

Que Monsieur F.ROBERT reçoive mes sincères remerciements pour ses enseignements et pour son amabilité d'avoir bien voulu participer au jury.

Je remercie également tous les membres du laboratoire TIM3 et en particulier J.CHAARANI, A.CHALABI, M.MERRAKCHI, M.MOUZALI, A.KERAGUEL et A.YASSINE pour leur contribution que chacun d'eux a pu apporter et l'équipe du service de reprographie pour l'excellente qualité de son travail.

Que tous mes amis et collègues qui ont rendu mon séjour en France agréable soient vivement remerciés : A.BEKKA, N.HAKIM, M.MELAL, M.LASSOUD, A.MESHKEH, M.MESSAOUD DJEBARA, H.TALAAAT et Z.ZITOUNI.



TABLES DES MATIERES

INTRODUCTION GENERALE.....	1
 CHAPITRE I LES METHODES EXISTANTES POUR LA RESOLUTION DES PROBLEMES DE REGRESSION EN ANALYSE DES DONNEES:	
<i>I.1. Introduction.....</i>	<i>9</i>
<i>I.2. Position du problème de la régression isotone relativement à un ordre total et sa résolution numérique.....</i>	<i>11</i>
<i>I.3. Position du problème de la régression isotone relativement à un ordre partiel et sa résolution numérique.....</i>	<i>19</i>
<i>I.4. Méthode de DYKSTRA pour les problèmes de régression.....</i>	<i>23</i>
<i>I.5. Applications de la méthode de DYKSTRA</i>	<i>26</i>
 CHAPITRE II ETUDE ET ADAPTATION DE LA METHODE DE L'INVERSE PARTIEL A LA RESOLUTION DES PROBLEMES DE REGRESSION :	
<i>II.1. Introduction.....</i>	<i>33</i>
<i>II.2. Rappels sur les opérateurs monotones maximaux.....</i>	<i>34</i>
<i>II.3. Minimisation d'une fonction convexe sur une intersection de convexes fermés</i>	<i>36</i>
<i>II.4. Minimisation d'une fonction quadratique sur une intersection de convexes fermés</i>	<i>43</i>
<i>II.5. Une méthode nouvelle de construction de la "Projection sur une intersection de convexes fermés.....</i>	<i>51</i>
<i>II.6. Dualité de FENCHEL pour le problème (4.1)-(4.2).....</i>	<i>53</i>
<i>II.7. Correspondance entre l'inverse partiel et les multiplicateurs de Lagrange.....</i>	<i>57</i>
<i>II.8. Applications.....</i>	<i>59</i>

CHAPITRE III ADAPTATION DE LA METHODE DE PIVOTAGE COMPLEMENTAIRE DE LEMKE A LA RESOLUTION DES PROBLEMES DE REGRESSION :

III.1. Introduction.....	71
III.2. Rappel sur la programmation linéaire.....	72
III.3. Programmation linéaire complémentaire et sa résolution numérique.....	74
III.4. Programmation quadratique et sa réduction en problème PLC.....	80
III.5. Applications.....	83

CHAPITRE IV RESOLUTION DES PROBLEMES DE REGRESSION PAR LA METHODE DU GRADIENT CONJUGUE :

IV.1. Introduction.....	87
IV.2. Problème de programmation quadratique et sa résolution numérique.....	88
IV.3. Dualité et fonction de Lagrange.....	106
IV.4. Applications.....	113

CHAPITRE V PRESENTATION DE SYNTHESE DES METHODES DE PROJECTIONS

V.1. Introduction.....	121
V.2. Méthode du gradient projeté pour la recherche de l'intersection ou de la distance de deux convexes fermés.....	124
V.3. Méthode de décomposition pour la recherche de l'intersection ou de la distance de deux convexes fermés.....	132
V.4. Méthode du gradient projeté relativement à une intersection convexes fermés.....	136
V.5. Méthode de décomposition pour la recherche de l'intersection de convexes fermés.....	141
V.6. Méthode du sous gradient pour la recherche de l'intersection de convexes fermés.....	149

V.7.	<i>Algorithme du minimax pour la recherche de l'intersection de convexes fermés.....</i>	157
V.8.	<i>Résumé général.....</i>	159
V.9.	<i>Recherche de l'intersection d'une famille de convexes fermés par croisement de méthodes de descente.....</i>	161
V.10	<i>Un survol des méthodes de projections.....</i>	168
V.11	<i>Classification des méthodes de projections.....</i>	179

CHAPITRE VI EXPERIMENTATIONS NUMERIQUES

VI.1	<i>Expérimentations numériques.....</i>	187
CONCLUSION GENERALE		199



INTRODUCTION GENERALE



L'omniprésence de la modélisation mathématique, tant dans le domaine des sciences de la terre que dans celui des sciences sociales, conduit à un grand nombre de problèmes d'analyse numérique s'exprimant en termes d'optimisation, et leur résolution relève des méthodes développées dans cette discipline. C'est en particulier le cas pour une grande classe de problèmes, résultant de la statistique ou plus généralement de l'analyse des données, dont l'importance pratique résulte de leur vaste champ d'applications.

Notre travail s'intéresse plus spécialement aux problèmes numériques posés par la régression isotone et la régression concave, dont l'intérêt concret est largement motivé par le développement récent des méthodes connues en psychométrie sous l'intitulé anglais de Multidimensional Scaling (MDS). Malgré une attention soutenue, marquée par les efforts considérables développés pour leur résolution numérique et le volume important de publications parues dans ce domaine depuis 1964, ces problèmes demeurent incomplètement résolus.

L'objectif de ce travail est de proposer des méthodes numériques performantes adaptées à ce type de problèmes et permettant la mise au point de logiciels répondant aux besoins des analystes de données.

L'apport de ce travail aux outils numériques utilisés par les statisticiens est double :

tout d'abord nous donnons une formulation générale des problèmes d'optimisation sous contraintes linéaires de type inégalité, rencontrés dans la mise en œuvre de techniques de régression aux moindres carrés. Nous proposons alors des algorithmes de résolution dont les propriétés de convergence sont connues, contrairement aux méthodes actuelles provenant des statisticiens. Nous montrons de plus que ces algorithmes étendent de façon notable la classe de problèmes statistiques que l'on sait résoudre.

Ainsi, ce travail permet-il de traiter les problèmes de régression isotone caractérisés par une contrainte de monotonie, par rapport à des relations d'ordre faible (en particulier d'ordre partiel), alors que les algorithmes actuels sont limités au cas des relations d'ordre ou préordre total, il répond donc à une demande forte en psychométrie et en économétrie. De même les problèmes de régression convexe/ concave obtiennent enfin une résolution algorithmique correcte. Ajoutons de plus que les problèmes d'optimisation sans contraintes de positivité, etc..., relèvent de cette approche.

En deuxième lieu, nous établissons une relation entre les approches numériques de type "gradient.." et les méthodes de projection sur l'intersection de cônes.

Le plan de ce mémoire est organisé en deux parties. La première partie, constituée des chapitres I à IV, est consacrée à la modélisation et la résolution numérique des problèmes de régression sous contraintes linéaires de type inégalités. Dans la seconde partie, au chapitre V, nous présentons une synthèse des méthodes de projections et nous mettons en évidence les liens qui les rattachent aux méthodes de type sous-gradient.

Dans le premier chapitre, nous présentons tout d'abord la classe des problèmes d'optimisation convexe rencontrés en analyse des données dans le cadre de la méthodologie de la régression isotone, de la régression concave, etc..... La méthodologie de la régression isotone est appliquée dans la résolution d'un grand nombre de problèmes statistiques caractérisés par l'existence d'informations ordinales, connues à priori, et prises en compte dans les phases d'estimation des paramètres du modèle, et de test d'hypothèses.

Elle forme en particulier la base des techniques modernes d'analyse des données dites de codage optimal, dont l'exemple le plus utilisé est l'analyse ordinales des tableaux de dissimilarités, connue sous l'intitulé anglais de Multidimensional Scaling.

La fin de ce chapitre est consacrée à la description des algorithmes usuels proposés par les analystes de données. Nous prenons alors soin de souligner les avantages et les inconvénients de ces algorithmes.

L'application des résultats de la théorie de la dualité nous permet alors de proposer trois méthodes pour résoudre ces problèmes.

- 1) La méthode de l'inverse partiel.
- 2) La méthode de pivotage complémentaire.
- 3) La méthode du gradient conjugué.

Chacun des chapitres suivant est consacré à la présentation d'une de ces méthodes et à son application au problème traité.

Ainsi, le chapitre II étudie la méthode de l'inverse partiel appliquée à un opérateur monotone maximal. Après avoir rappelé les définitions et les propriétés fondamentales d'un opérateur monotone maximal, dûs à G.J.MINTY, J.J.MOREAU, R.T.ROCKAFELLAR, F.BROWDER, H.BREZIS, nous décrivons l'algorithme général de l'inverse partiel, qui est étudié notamment par J.J.MOREAU, R.T.ROCKAFELLAR et J.E.SPINGARN.

Nous exposons alors la technique de plongement dans l'espace produit, qui est reprise dans le cinquième chapitre lors de l'étude des méthodes de projection, et nous illustrons le champ d'application de cet algorithme.

Nous montrons alors comment l'algorithme de l'inverse partiel, appliqué à la minimisation d'une fonction convexe sur l'intersection de cônes convexes C_i (type de contraintes usuel en analyse des données ordinales) permet aussi de résoudre le problème dual résultant de la dualité de W.FENCHEL, associée à la notion de perturbation horizontale. Un résultat équivalent est obtenu dans le cadre de la dualité classique à l'aide des techniques de la perturbation verticale.

Nous explicitons alors la modélisation des problèmes de régression isotone et de régression concave dans ce cadre, et nous exposons la version de cet algorithme, qui devient très simplifiée, adaptée au traitement de ces deux cas.

Le chapitre III est consacré à la méthode de pivotage complémentaire ou algorithme de C.LEMKE. On peut considérer cette approche comme une adaptation de la méthode du simplexe à la résolution de la classe des problèmes linéaires complémentaires (PLC), dont les programmes quadratiques forment une sous classe. Nous montrons alors que l'algorithme de C.LEMKE, qui converge en un nombre fini d'itérations, permet de résoudre simultanément les problèmes primal et dual dans le cas de la programmation quadratique.

La compréhension de cet algorithme est facilitée par la présentation d'exemples simples.

L'application de la méthode classique du gradient conjugué à la minimisation d'une forme quadratique convexe est présentée dans la première partie du chapitre IV. Nous montrons alors comment on peut utiliser la méthode du gradient conjugué pour la minimisation d'une forme quadratique convexe sous des contraintes linéaires, c'est à dire en programmation quadratique, et nous décrivons la forme appropriée de l'algorithme, qui converge en un nombre fini d'itérations, connue sous le nom de gradient conjugué conditionnel.

On présentera le problème dual classique, via la perturbation verticale, puis discutera des avantages et inconvénients de l'application de la méthode du gradient conjugué conditionnel aux problèmes primal et dual.

Le chapitre V propose une étude synthétique des méthodes de projection actuellement connues. Le champ d'application de ces méthodes dépasse le cadre de l'analyse des données, puisqu'elles peuvent être appliquées pour résoudre des programmes linéaires, et des systèmes d'égalités ou d'inégalités linéaires.

Pour résoudre le problème de la recherche d'un point commun à une famille de convexes fermés, la plupart des méthodes proposées, que ce soit les méthodes barycentriques basées sur le procédé de G.CIMMINO ou les méthodes de projection successives issues des travaux de S.KACZMARZ, sont fondées sur la notion de projection, ce qui nous a conduit à rechercher un point de vue unificateur, et justifie par-là même la longueur de ce chapitre.

Nous établissons en particulier les relations liant les méthodes de projections existantes, ainsi que leur classification, puis nous présentons des méthodes nouvelles de projection, fondées sur la représentation du problème dans un espace produit. Nous montrons enfin que l'ensemble des méthodes présentées s'inscrivent dans celles du type sous-gradient.

Le dernier chapitre est consacré à la présentation et à la comparaison des résultats numériques donnés par les méthodes étudiées dans les chapitres I, II, III et IV.

CHAPITRE I

***LES METHODES EXISTANTES POUR LA RESOLUTION DES
PROBLEMES DE REGRESSION EN ANALYSES DES DONNEES .***



1.1. INTRODUCTION :

La méthodologie de la régression isotone, est appliquée dans la résolution d'un grand nombre de problèmes statistiques caractérisés par l'existence d'informations ordinales, connues à priori, et prises en compte dans les phases d'estimation des paramètres du modèle et de test d'hypothèses. Dans le cadre de l'Analyse des Données, cette méthodologie forme la base des techniques de codage optimal, dont l'exemple le plus utilisé est connu sous l'intitulé anglais de *Multidimensional Scaling* (cf R.N.SHEPARD[15], J.KRUSKAL[12]).

Sous sa forme la plus simple, la régression isotone traite le problème statistique suivant. On considère la structure statistique associée à la comparaison de n populations, à l'aide d'échantillons de taille respective η_i .

$\prod_{i=1}^n (\mathbb{R}^n, \mathcal{B}_{\mathbb{R}}, P_i) * \eta_i$, où P_i est une loi de Gauss de moyenne μ_i et de variance connue σ_i .

On suppose de plus donnée une relation d'ordre entre populations, caractérisée par

$$\mu_i \leq \mu_{i+1} \text{ pour } i=1, \dots, n-1.$$

On se propose alors d'estimer ces paramètres μ_i , au sens du maximum de vraisemblance, on recherche alors le maximum en μ de l'équation :

$$\prod_{i=1}^n \frac{\sqrt{\eta_i}}{\sqrt{2\pi} \sigma_i} \text{Exp} \left[\frac{-\eta_i}{2} \frac{(g_i - \mu_i)^2}{\sigma_i^2} \right].$$

sous les contraintes

$$\mu_i \leq \mu_{i+1} \text{ pour } i=1, \dots, n-1.$$

Ce problème est équivalent au problème de régression, au sens des moindres carrés, suivant noté par la suite (P_1) :

$$\min \sum_{i=1}^n w_i (g_i - \mu_i)^2$$

sous les contraintes

$$\mu_i \leq \mu_{i+1} \text{ pour } i=1, \dots, n-1.$$

qui traduisent le fait que $\mu = (\mu_1, \mu_2, \dots, \mu_n)$ est une fonction isotone sur $A = \{1, 2, \dots, n\}$, muni de la relation d'ordre naturel (α) sur les entiers..

L'extention de cette classe de problèmes au cas de contraintes de concavité est liée aux besoins de la modélisation statistique en économétrie.

Ainsi, le problème de la régression concave, noté par la suite (P_2), consiste à minimiser

$$\sum_{i=1}^n w_i (g_i - x_i)^2$$

sous les contraintes de concavités

$$\frac{x_{i+2} - x_{i+1}}{y_{i+2} - y_{i+1}} \leq \frac{x_{i+1} - x_i}{y_{i+1} - y_i} ; i=1,2,\dots,n-2.$$

Où les variables $y_i, i=1,\dots,n$, sont telles que $0 < y_1 < y_2 < \dots < y_n$.

Ce chapitre expose, en un premier temps, les problèmes de la *régression isotone*, et de la *régression concave* tels qu'ils sont posés en *Analyse des Données*. La structure particulière de ces problèmes a permis l'élaboration d'un grand nombre d'algorithmes, citons entre autres J.KRUSKAL [12] et [13], W.A.THOMPSON [16], Chien-Fu WU [19], J.R.CHOI [6], R.DYKSTRA [7] et [8], et M. BOUGHAZI et al [3].

On présente ensuite l'algorithme de J.KRUSKAL destiné à résoudre le problème de régression isotone relativement à un ordre total ainsi que l'algorithme de W.A.THOMPSON pour un ordre partiel. L'algorithme de R.DYKSTRA, pour la résolution des problèmes de la régression isotone relativement à un ordre quelconque et celui de la régression concave, est passé en revue. La technique développée par R.DYKSTRA, bien qu'elle résolve une grande classe de problèmes de régression, présente l'inconvénient de converger très lentement et même parfois de boucler.

Il convient de signaler que la méthode de J.KRUSKAL ou *Pool Adjacent Violators Algorithm (PAV)*, qui est simple et rapide, n'est applicable qu'au problème de la *régression isotone* relativement à un ordre total. En revanche, le problème de la *régression isotone* pour le cas général ainsi que le problème de la *régression concave* restent sans solutions satisfaisantes, malgré les tentatives d'extension du PAV (cf W.A.THOMPSON [16], Chien-Fu WU [19], J.R.CHOI [6]) afin de parvenir à une solution rigoureuse dans le cas d'un ordre partiel.

On se limite à ces méthodes du fait de leur intérêt pratique remarquable (la méthode J.KRUSKAL et celle de W.A.THOMPSON) et théorique (l'algorithme de R.DYKSTRA).

1.2. POSITION DU PROBLEME DE LA REGRESSION ISOTONE RELATIVEMENT A UNE RELATION D'ORDRE TOTAL :

DEFINITIONS :

Soit $X = \{x_1, x_2, \dots, x_n\}$ un ensemble muni d'une relation d'ordre total notée α , telle que : $x_1 \alpha x_2 \alpha \dots \alpha x_n$.

Une application f à valeurs réelles, définie sur X , est dite *isotone* si elle vérifie :

$$\forall (x_k, x_{k'}) \in X^2 : x_k \alpha x_{k'} \Rightarrow f(x_k) \leq f(x_{k'}) .$$

On appellera I_X l'ensemble des applications *isotones* sur X .

Soit g une application à valeurs réelles, définie sur X et w une application à valeurs positives, également définie sur X (w définit un ensemble de poids).

On appelle *régression isotone* de g pour la relation d'ordre α , relativement aux poids w , une application g^* de I_X vérifiant :

$$\forall f \in I_X : \sum_{i=1}^n w_i (g_i - g_i^*)^2 \leq \sum_{i=1}^n w_i (g_i - f_i)^2 .$$

(pour toute application définie sur X , on pose : $g(x_k) = g_k$).

Ainsi, le problème de *régression isotone*, noté (P_1) , consiste à minimiser

$$\sum_{i=1}^n w_i (g_i - \mu_i)^2$$

sous les contraintes

$$\mu_i - \mu_{i+1} \leq 0 \quad i=1, 2, \dots, n-1.$$

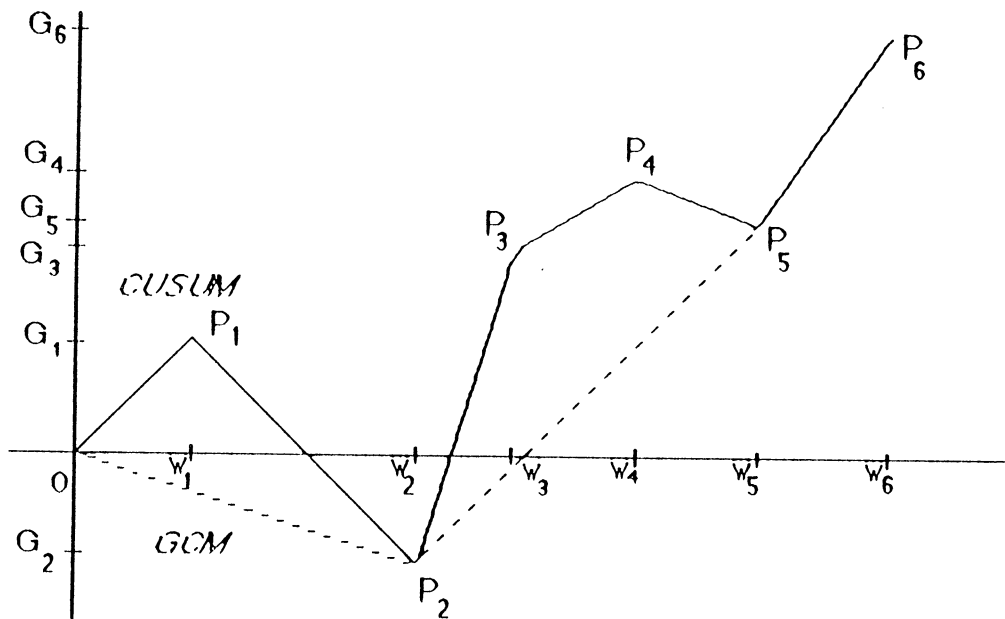
1.2.1. INTERPRETATION GRAPHIQUE :

Dans un repère orthonormé du plan, on considère n points tels que

$$P_k = (W_k, G_k) , \text{ où } W_k = \sum_{i=1}^k w_i , \text{ et } G_k = \sum_{i=1}^k w_i g_i ;$$

Où g et w sont les deux applications réelles définies ci-dessus. On considère de plus le point $P_0 = (w_0, G_0) = (0, 0)$.

Le diagramme obtenu par interpolation linéaire entre les points P_k est appelé "*Diagramme des Sommes Cumulées*" (*Cumulative Sum Diagram* noté **CUSUM**).



CUSUM : *Diagramme des Sommes Cumulées*
 et **GCM** : *Plus Grand Minorant Convexe du CSD.*

La pente du segment joignant les points P_{k-1} et $P_{k'}$ ($k \leq k'$) sur le **CUSUM** est la *moyenne pondérée* qu'on désigne par **Mp** :

$$Mp(x_k, \dots, x_{k'}) = \frac{\sum_{i=k}^{k'} w_i g_i}{\sum_{i=k}^{k'} w_i} ;$$

qui vaut g_k si $k=k'$.

On est alors amené à considérer le "*Plus Grand Minorant Convexe*" (*Greatest Convex Minorant*=**GCM**) du **CUSUM**, c'est à dire l'enveloppe convexe inférieure des points du **CUSUM**, dont les points P_k^* d'abscisse w_k , $k=0,1,\dots,n$, ont pour ordonnée G_k^* .

On montre le théorème suivant :

THEOREME 1 [1]:

En chaque point de X, la pente g^ de GCM est la régression isotone de g , et elle est unique.*

On peut également montrer la propriété suivante [1] :

PROPRIETE : Soient $1 \leq j \leq k \leq n$, tels que : $g_j^* = g_{j+1}^* = \dots = g_k^*$, avec :

$$g_{j-1}^* \neq g_j^*, \text{ et } g_k^* \neq g_{k+1}^*$$

(cette dernière inégalité seulement si $k \neq n$) ; on a alors :

$$g_j^* = \dots = g_k^* = \frac{\sum_{i=j}^k w_i g_i}{\sum_{i=j}^k w_i}$$

Et l'on remarque de plus que :

$$\forall k \in \{1, 2, \dots, n\} : g_{k-1} \geq g_k \Rightarrow g_{k-1}^* = g_k^*$$

Cette propriété fonde l'algorithme du "POOL ADJACENT VIOLATORS" que l'on illustre dans le paragraphe qui suit.

Remarque : On peut montrer [1] que pour tout réel μ , et pour tout réel positif λ , on a en chaque point de X : $(g+\mu)^* = g^* + \mu$, et : $(\lambda g)^* = \lambda g^*$.

1.2.2. EXEMPLE DE REGRESSION ISOTONE :

Supposons que le cardinal de X est 6, et qu'une application réelle g sur X est définie par les six valeurs :

$g_1 = g(x_1) = 2$; $g_2 = 6$; $g_3 = 2$; $g_4 = 13$; $g_5 = 7$; $g_6 = 8$; prenons pour poids respectifs :

$$w_1 = w(x_1) = 2$$
 ; $w_2 = 1$; $w_3 = 3$; $w_4 = 1$; $w_5 = 2$; $w_6 = 3$.

La *régression isotone* de g s'obtient alors en calculant la moyenne pondérée des "blocs" de valeurs consécutives non croissantes, de la façon suivante :

w_k	g_k	g'_k	g''_k	g_k^*
2	2	2	2	2
1	6	3	3	3
3	2			
1	13	7	9	8.5
2	7			
3	8	8	8	8.5

g^* , application isotone définie par les six valeurs g_k^* , constitue la *régression isotone* de g pour la relation d'ordre définie sur X , relativement aux poids w_k . La technique utilisée ci-dessus est celle employée par l'algorithme **U.D.B** décrit ci-dessous..

1.2.3. ALGORITHME DE "POOL ADJACENT VIOLATORS" :

Il est clair que si, pour un i quelconque, on a $g_{i-1} > g_i$, alors le graphe de la partie du **GCM** entre les points P_{i-2}^* et P_i^* est un segment de droite. Ainsi, le **CUSUM** peut être modifié en joignant P_{i-2} à P_i par un segment sans changer le **GCM**. Cette méthode d'approximations successives du **GCM** peut être décrite algébriquement comme l'algorithme du regroupement des violateurs adjacents (*Pool Adjacents Violators Algorithm*, ou *PAV algorithm*).

La *régression isotone* g^* partitionne X en sous-ensembles ou blocs sur lesquels elle prend une valeur constante, appelé *bloc solution*. Sur chacun de ces blocs solution la valeur de g^* est la moyenne pondérée des valeurs de g sur le chaque bloc, i.e. la pente de la partie de **GCM** correspondant à ce bloc. Ainsi, cela suffit pour trouver les *blocs solution*. En décrivant l'algorithme, un ensemble arbitraire d'éléments consécutifs de X est pris comme bloc de référence.

L'algorithme commence avec le partitionnement le plus fin possible, à savoir les points de X , et on regroupe les blocs étape par étape jusqu'à l'obtention de la partition finale.

Si $g_1 \leq g_2 \leq \dots \leq g_n$, alors la partition initiale est la partition finale et $g_i^* = g_i$, $i=1, \dots, n$. Sinon, on choisit n'importe quelle paire des violateurs, c'est à dire un i tel que $g_i > g_{i+1}$ puis on regroupe ces deux valeurs de g , i.e. les deux points x_i et x_{i+1} en un seul bloc $\{x_i, x_{i+1}\}$ auquel on associe la valeur moyenne :

$$Pm(x_i, x_{i+1}) = \frac{w_i g_i + w_{i+1} g_{i+1}}{w_i + w_{i+1}}$$

et le poids $w_i + w_{i+1}$.

Après chaque itération de l'algorithme, la valeur des moyennes associées aux blocs sont examinées pour voir si elles respectent ou non la relation d'ordre donnée. Si cela est, la partition finale est obtenue, et la valeur de g^* en chaque point d'un bloc est la valeur moyenne associée au bloc. Sinon, une paire de violateurs adjacents est sélectionnée, et regroupée pour former un nouveau bloc, auquel on associe la valeur moyenne et la somme des poids. Puis on procède à une nouvelle itération de l'algorithme.

1.2.4. ALGORITHME "UP-DOWN BLOCKS" DE KRUSKAL :

L'algorithme "UP-DOWN BLOCKS" noté U.D.B développé par J.P.KRUSKAL est une variante de l'algorithme P.A.V décrit ci-dessus et appliqué au cas où la relation d'ordre est totale. Les éléments de X , rangés dans un ordre arbitraire, sont notés par x_1, x_2, \dots, x_n .

Le terme de bloc est utilisé pour désigner un ensemble d'éléments consécutifs de X . Si B_-, B, B_+ sont 3 blocs consécutifs dans l'ordre, on dit que B est *up-satisfied* si $MpB < MpB_+$, MpB désigne la moyenne pondérée associée au bloc B , B est dit *down-satisfied* si la $MpB_- < MpB$.

Il est supposé que tout bloc contenant $\{x_1\}$ est *down-satisfied* et tout bloc contenant $\{x_n\}$ est *up-satisfied*.

L'algorithme commence avec une partition de X en blocs ne contenant qu'un seul élément et on procède à un regroupement de blocs jusqu'à l'obtention de la solution. La valeur de g^* est alors donnée par les moyennes pondérées associées aux blocs de la partition finale.

A chaque itération de l'algorithme un seul bloc est actif. Il peut être regroupé avec un bloc adjacent ou bien, s'il est *up-satisfied* et *down-satisfied*, alors le bloc adjacent suivant devient le bloc actif.

Le principe de l'algorithme est expliquée par le schéma ci-dessous dans lequel figurent les abréviations suivantes :

US : le bloc actif est *up-satisfied*.

DS : le bloc actif est *down-satisfied*.

RU : regrouper le bloc actif avec le bloc supérieur suivant, le nouveau bloc devient actif.

RD : regrouper le bloc actif et le bloc précédent, le bloc devient obtenu devient le bloc actif.

BS : le bloc suivant devient le bloc actif.

ORGANIGRAMME DE L'ALGORITHME DE KRUSKAL.

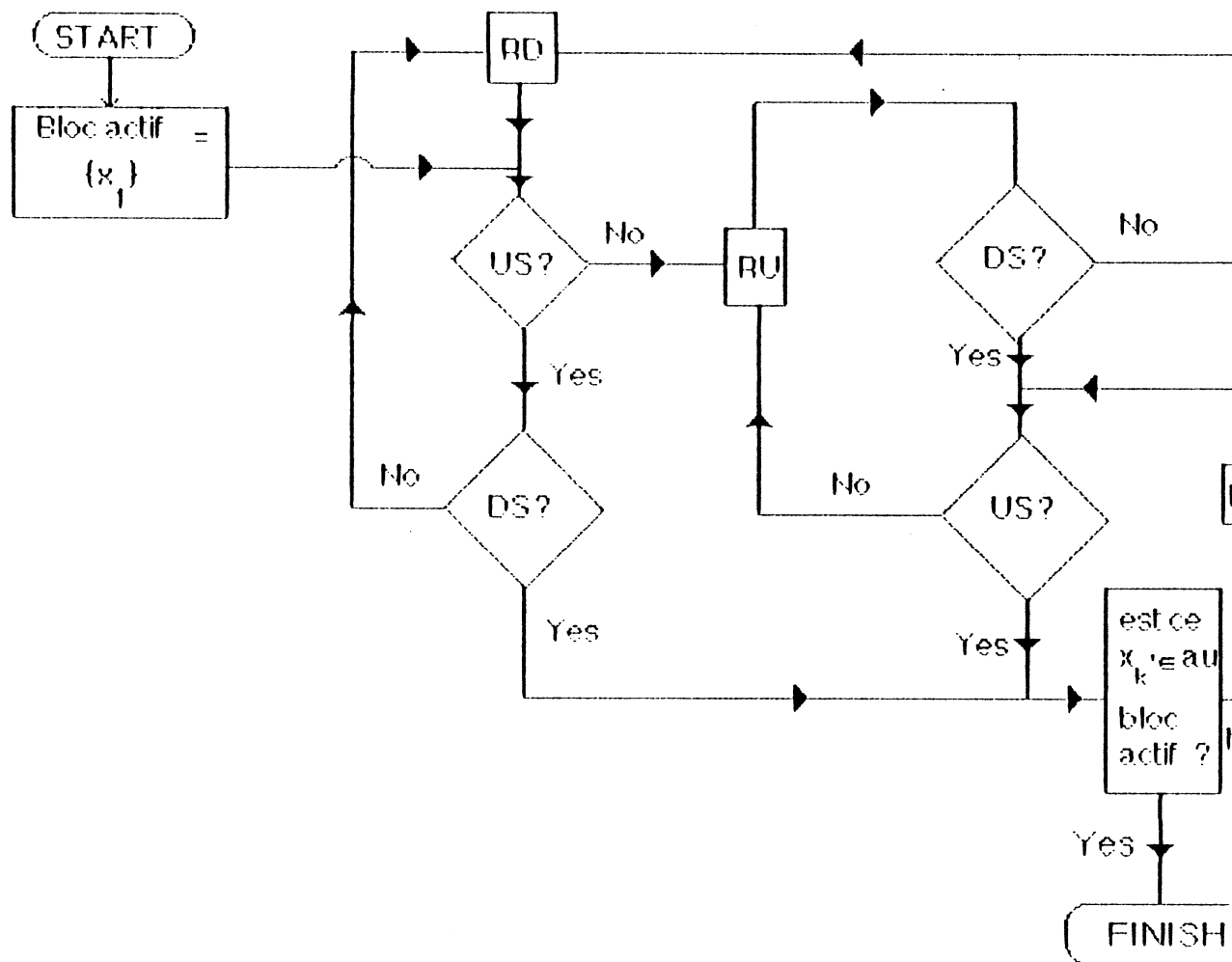


Figure (3). Schéma de l'algorithme de Kruskal "Up-and-Down Blocks".

1.2.5. LE PROBLEME DE CODAGE MULTIDIMENSIONNEL OPTIMAL; UNE APPLICATION DE LA REGRESSION ISOTONE RELATIVEMENT A UN ORDRE TOTAL

Etant donnés n objets observés o_1, o_2, \dots, o_n qui, dans le contexte psychologique, sont souvent appelés des *stimuli*. Soit δ_{ij} la mesure de la dissimilarité entre l'objet o_i et l'objet o_j . δ_{ij} est le terme général d'une matrice carrée Δ d'ordre n qui est la donnée du problème de codage multidimensionnel optimal. De plus, on suppose donnée une matrice W d'ordre n dont les éléments sont interprétés comme des poids; w_{ij} indique la précision de mesure de δ_{ij} où la probabilité d'apparition de la valeur δ_{ij} .

L'objectif d'une méthode de codage multidimensionnel optimal est de représenter les objets comme des points dans un espace métrique (H, d) de dimension fixée par l'utilisateur, de sorte que les distances $d(x_i, x_j)$ soient approximativement égales aux dissimilarités δ_{ij} . On écrit souvent d_{ij} ou $d_{ij}(x)$ au lieu de $d(x_i, x_j)$. La relation isotone qu'on cherche est formulée comme suit. Soit X^2 l'ensemble des tous les points (i, j) , $1 \leq i < j \leq n$. On définit un ordre total " α " sur X^2 par :

$$(i_1, j_1) \alpha (i_2, j_2) \text{ si } \delta_{i_1 j_1} \leq \delta_{i_2 j_2}$$

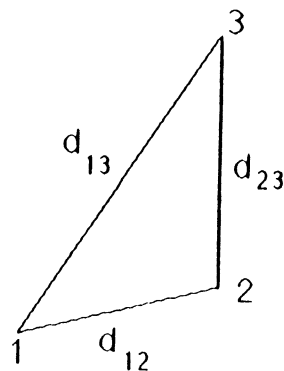
Où δ_{ij} désigne la dissimilarité entre l'objet o_i et l'objet o_j . On se propose alors de construire n points dans l'espace H de sorte que

$$(i_1, j_1) \alpha (i_2, j_2) \Rightarrow d_{i_1 j_1} \leq d_{i_2 j_2}$$

Une telle distance est une application isotone sur l'ensemble X^2 relativement à l'ordre total " α ".

EXEMPLE 1: Un exemple est donné par le tableau et la figure ci-dessous

	x_1	x_2	x_3
x_1	0	2	5
x_2	2	0	3
x_3	5	3	0



Ici on a choisi $n=3$ et il est possible de trouver 3 points dans l'espace \mathbb{R}^2 de sorte que leurs distances respectent la relation d'ordre définie par les dissimilarités données.

$$d_{12} \leq d_{23} \leq d_{13} \quad ; \quad \delta_{12} \leq \delta_{23} \leq \delta_{13}$$

EXEMPLE 2 : Etant donné 4 objets dont les dissimilarités mesurées entre ces objets vérifie la relation d'ordre suivante :

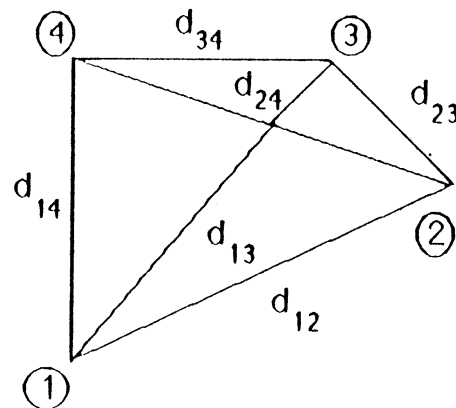
$$\delta_{23} \leq \delta_{34} \leq \delta_{14} \leq \delta_{13} \leq \delta_{24} \leq \delta_{12}$$

On a pu trouver 4 points dans l'espace \mathbb{R}^2 de sorte que leurs distances respectent la relation définie sur les dissimilarités, i.e.

$$d_{23} \leq d_{34} \leq d_{14} \leq d_{13} \leq d_{24} \leq d_{12}$$

Le tableau et la figure suivant illustre cette possibilité.

	x_1	x_2	x_3	x_4
x_1	0	$\sqrt{29}$	5	4
x_2	$\sqrt{29}$	0	2	5.2
x_3	5	2	0	3
x_4	4	5.2	3	0



Cependant, dans d'autres situations on peut ne pas trouver de tels points.

EXEMPLE 3 : Etant donné 4 objets dont les dissimilarités mesurées entre ces objets vérifie la relation d'ordre suivante :

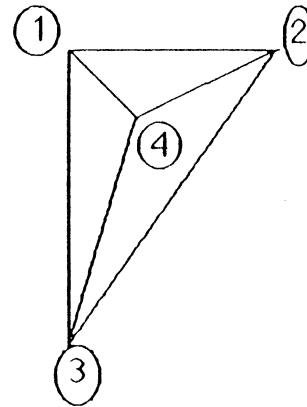
$$\delta_{12} \leq \delta_{13} \leq \delta_{23} \leq \delta_{14} \quad \text{et} \quad \delta_{24} \leq \delta_{34} \leq \delta_{12}$$

On ne peut pas trouver 4 points dans l'espace \mathbb{R}^2 de sorte que leurs distances respectent la relation définie sur les dissimilarités, i.e.

$$d_{12} \leq d_{13} \leq d_{23} \leq d_{14} \quad \text{et} \quad d_{24} \leq d_{34} \leq d_{12}$$

Le tableau et la figure suivant illustre cette impossibilité.

	x_1	x_2	x_3	x_4
x_1	0	3	4	1
x_2	3	0	5	2
x_3	4	5	0	2.5
x_4	1	2	2.5	0



Soit d la régression isotone de d sur X^2 .

Le problème posé en *codage multidimensionnel optimal* se formule ainsi :

$$\min \sum_{i=1}^n \sum_{j=1}^{i-1} w_{ij} (d_{ij} - d_{ij})^2$$

sous les contraintes

$$d_{i_1, j_1} \leq d_{i_2, j_2} \quad \text{si } (i_1, j_1) \alpha (i_2, j_2).$$

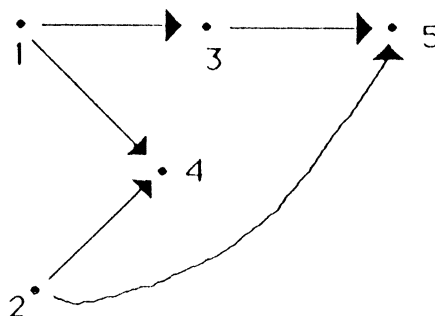
1.3. REGRESSION ISOTONE RELATIVEMENT A UNE RELATION D'ORDRE PARTIEL :

W.A. THOMPSON [16] a proposé un algorithme pour un cas plus général, où la relation l'ordre est partiel. Il propose de plus que chaque élément de X , à l'exception du premier élément x_1 , a exactement un prédécesseur immédiat.

DEFINITION 1 :

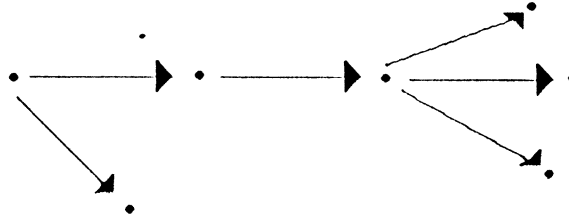
Un élément x de X est un prédécesseur immédiat d'un élément y si dans l'ordre partiel " \leq " $x \leq y$ mais il n'existe aucun $z \in X$ différent de x et y tel que $x \leq z \leq y$.

Exemple 1 : Soit une relation d'ordre " \leq " définie sur $X = \{x_1, x_2, x_3, x_4, x_5\}$ par $x_1 \leq x_3 \leq x_5$, $x_1 \leq x_4$, $x_2 \leq x_4$, $x_2 \leq x_5$, dont le graphe est représenté sur la figure ci-dessous :



Dans cette relation, x_4 a deux prédécesseurs immédiats x_1 et x_2 , x_5 a deux prédécesseurs immédiats x_2 et x_3 , x_3 a un seul prédécesseur immédiat et x_1 et x_2 n'ont pas de prédécesseur immédiat.

Les relations d'ordre pour lesquels l'algorithme de W.A.THOMPSON s'applique sont représentés par un arbre. Dans ce cas tout élément de X , à l'exception du premier (la racine) admet un prédécesseur immédiat. La figure suivante illustre le cas.



1.3.1. DESCRIPTION DE L'ALGORITHME :

L'algorithme de W.A.THOMPSON est applicable dans le cas où chaque élément, à l'exception de la racine, a exactement un prédécesseur immédiat. Cet algorithme commence comme dans la méthode de J.KRUSKAL avec la partition la plus fine, à savoir les points de X .

On dit que y est un violateur si $g(y) < g(x)$ où x est un prédécesseur immédiat de y .

On dit que y est un violateur minimal si la valeur de g en ce point est minimale par rapport à ses valeurs sur les autres violateurs.

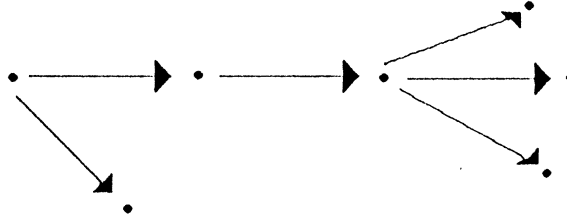
Après l'étape de partitionnement, l'algorithme procède en sélectionnant un violateur minimal que l'on regroupe à son prédécesseur immédiat pour former un bloc. A une itération donnée de l'algorithme, l'ensemble X est partitionné en blocs. Chaque bloc a pour poids, la somme des poids de chacun de ses éléments, pour valeur, la moyenne pondérée des valeurs de g en chacun de ses points et pour racine le point du bloc dont le prédécesseur immédiat n'appartient au bloc.

Un prédécesseur immédiat d'un bloc est tout bloc contenant le prédécesseur immédiat de la racine du bloc considéré. Quand un bloc et son prédécesseur immédiat sont regroupés en un seul bloc, la racine du nouveau bloc est la racine du bloc prédécesseur immédiat.

Chaque étape de l'algorithme consiste à regrouper un bloc violateur minimal avec son prédécesseur immédiat. L'algorithme continue ainsi jusqu'à ne plus avoir de violateurs. A ce point, les blocs ont pour valeurs les moyennes pondérées des points qui les constituent et la valeur de g^* en chaque point est la valeur du bloc auquel appartient ce point.

1.3.2. EXEMPLE : Etant donné $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ on définit une relation d'ordre " α " sur X par :

$x_1 \alpha x_2 \alpha x_3 \alpha x_4, x_1 \alpha x_7, x_3 \alpha x_5, x_3 \alpha x_6$. Le schéma représentatif de cette relation est comme suit :



On remarque que cette relation vérifie le critère proposé plus haut. Chaque x_i , à l'exception de x_1 , admet un seul prédécesseur immédiat.

Soit une application réelle g sur X définie par les sept valeurs :

$$g(x_1)=4, g(x_2)=7, g(x_3)=18, g(x_4)=20, g(x_5)=6, g(x_6)=-2, g(x_7)=2,$$

prenons pour poids :

$$w(x_1)=w(x_2)=w(x_3)=w(x_4)=w(x_5)=w(x_6)=w(x_7)=1.$$

La régression isotone g^* de g associée à X et l'ordre " α " doit vérifier les contraintes suivantes :

$$g^*(x_1) \leq g^*(x_2) \leq g^*(x_3) \leq g^*(x_4) ; g^*(x_3) \leq g^*(x_5) ; g^*(x_3) \leq g^*(x_6) ; g^*(x_1) \leq g^*(x_7).$$

On représente les données sous forme d'un tableau

x	x_1	x_2	x_3	x_4	x_5	x_6	x_7
w	1	1	1	1	1	1	1
g	4	7	18	20	6	-2	2

L'algorithme de W.A.THOMPSON pour le calcul de g^* procède ainsi :

Etape 1 : Les violateurs à cette étape sont (x_5, x_6, x_7) car

$$g(x_5) < g(x_3), g(x_6) < g(x_3) \text{ et } g(x_7) < g(x_1)$$

Le violeur minimal est x_6 puisque la valeur $g(x_6)=-2$ est inférieure à la valeur $g(x_5)=6$ et à la valeur $g(x_7)=2$. On regroupe alors x_3 et x_6 en un bloc $\{x_3, x_6\}$ de poids $1+1=2$ et de valeur $(18-2)/2=8$.

Et on aura le tableau correspondant

x	x_1	x_2	$\{x_3, x_6\}$	x_4	x_5	x_7
w	1	1	2	1	1	1
g	4	7	8	20	6	2

Etape 2 : A cette étape les violeurs sont x_5 et x_7 car

$$g(x_5) < g(x_3) \text{ et } g(x_7) < g(x_1).$$

Le violeur minimal est x_7 puisque la valeur $g(x_7)=2$ est inférieure à la valeur $g(x_5)=6$. On regroupe alors x_1 , x_7 et l'on obtient le bloc $\{x_1, x_7\}$ de poids $1+1=2$ et de valeur $(2+4)/2=3$.

On obtient ainsi le tableau correspondant

x	$\{x_1, x_7\}$	x_2	$\{x_3, x_6\}$	x_4	x_5
w	2	1	2	1	1
g	3	7	8	20	6

Etape 3 : Le seul bloc violeur à cette étape est le bloc $\{x_5\}$ car la valeur correspondante au bloc $\{x_5\}$ est strictement inférieure à celle du bloc $\{x_3, x_6\}$, donc le bloc $\{x_5\}$ est le bloc violeur minimal. En regroupant le bloc $\{x_3, x_6\}$ et le bloc $\{x_5\}$, on obtient le bloc $\{x_3, x_5, x_6\}$ de poids $1+2=3$ et de valeur $(16+6)/3=22/3$.

On construit alors le tableau correspondant

x	$\{x_1, x_7\}$	x_2	$\{x_3, x_5, x_6\}$	x_4
w	2	1	3	1
g	3	7	22/3	20

Etape 4 : On a plus de violateur, et l'algorithme s'arrête et on obtient la solution $g^*=(3, 9, 22/3, 20, 22/3, 22/3, 3)$ qui constitue la régression isotone de g relativement à l'ordre " α " défini sur X , relativement aux poids $w(x_k)$.

1.4. METHODE DE DYKSTRA :

En général, les problèmes de régression ont une structure particulière , ce qui a permis d'élaborer des algorithmes particulièrement simples et rapides pour certains cas, tels que, l'algorithme de J.KRUSKAL pour la régression isotone relativement à un ordre total, l'algorithme de W.A.THOMPSON pour la régression isotone relativement à un ordre partiel bien particulier. Cependant, le problème de la régression isotone relativement à un ordre partiel quelconque et celui de la régression concave restent sans solutions satisfaisantes, malgré les tentatives d'extention de la technique de l'algorithme "*POOL ADJACENT VIOLATORS*" (cf W.A.THOMPSON[16],C.F.WU[19] et V.EEDEN[10]). Et dans le but de construire une solution pour tous les problèmes de régression , R.DYKSTRA a interprété ces problèmes comme des problèmes aux moindres carrés sous des contraintes linéaires de types inégalités. Il a développé alors une technique, distincte de celle du "*POOL ADJACENT VIOLATORS ALGORITHM*", et qui exploite le fait que chaque contrainte définit un cône de structure simple. Le problème revient alors à chercher la projection d'un point sur l'intersection des cônes définis par l'ensemble des contraintes. R.DYKSTRA en formulant le problème ainsi, propose une méthode apparemment avantageuse qui consiste à ne faire intervenir à chaque itération qu'un seul cône. Par suite l'algorithme R.DYKSTRA itère les projections sur des cônes simples à décrire. On traite alors un problème facile à résoudre, jusqu'à l'obtention de la solution.

1.4.1. POSITION DU PROBLEME :

Soient g et w ($w_i > 0$) des éléments donnés de \mathbb{R}^n . On définit le produit scalaire de x et y (relativement aux poids w_i) comme suit

$$\langle x, y \rangle = \sum_{i=1}^n w_i x_i y_i$$

La norme associée à ce produit scalaire est donnée par

$$\|x\|^2 = \langle x, x \rangle = \sum_{i=1}^n w_i x_i^2$$

Et le problème consiste à minimiser

$$f(x) = 1/2 \|g - x\|^2 \quad (4.1)$$

Sous les contraintes

$$x \in \bigcap_{i=1}^m C_i \quad (4.2)$$

Où les C_1, C_2, \dots, C_m sont des cônes convexes fermés, dans \mathbb{R}^n .

1.4.2. DESCRIPTION DE L'ALGORITHME :

Definissons la projection de x sur C_i par $P(x/C_i)$, et Pour tout entier positif n , posons $n \bmod m = i$ si $n = kr + i$ avec $1 \leq i \leq m$.

Etape de l'initialisation : soit $g_0 = g$, $I_0 = 0$, $i = 1, 2, \dots, m$ et $n = 1$.

Etape (n) :

L'algorithme compose les projections sur les cônes C_i pris en séquence- ainsi :

1°) Soit $g_{1,1}$ représente la projection de g sur le cône C_1 . On pose $I_{1,1} = g_{1,1} - g$ qui désigne le facteur d'incrémententation.

2°) Soit $g_{1,2}$ la projection de $g + I_{1,1} = g_{1,1}$ sur C_2 , le facteur d'incrémententation devient alors $I_{1,2} = g_{1,2} - g_{1,1} = g_{1,2} - (g + I_{1,1})$ par suite on a :

$$g_{1,2} = g + I_{1,1} + I_{1,2}$$

3°) Soit $g_{1,3}$ la projection de $g_{1,2} = g + I_{1,1} + I_{1,2}$ sur C_3 , le facteur d'incrémententation devient

$$I_{1,3} = g_{1,3} - g_{1,2} = g_{1,3} - (g + I_{1,1} + I_{1,2})$$

Ce qui donne :

$$g_{1,3} = g + I_{1,1} + I_{1,2} + I_{1,3}.$$

4°) On continue jusqu'à l'obtention de $g_{1,m}$ et $I_{1,m} = g_{1,r} - (g + I_{1,1} + \dots + I_{1,m-1})$.

Soit $g_{2,1}$ la projection de $(g_{1,m} - I_{1,1})$ sur C_1 ,

$$(g_{1,m} - I_{1,1}) = (g + I_{1,2} + \dots + I_{1,m})$$

et donc

$$g_{2,1} = P(g + I_{1,2} + \dots + I_{1,m} / C_1)$$

$$I_{2,1} = g_{2,1} - g_{1,m} = g_{2,1} - (g + I_{1,2} + \dots + I_{1,m}).$$

$$g_{2,1} = g + I_{2,1} + I_{1,2} + \dots + I_{1,m}.$$

On remarque qu'en projetant cycliquement sur les cônes, $g_{n,i}$ est la projection de $g + I_{n,1} + \dots + I_{n,i-1} + I_{n-1,i+1} + \dots + I_{n-1,m}$ sur le $i^{\text{ème}}$ cône durant le $n^{\text{ième}}$ cycle. Ainsi, en général, $g_{n,i}$ est la projection de

$$g + I_{n,1} + \dots + I_{n,i-1} + I_{n-1,i+1} + \dots + I_{n-1,m}$$

sur le cône C_i et le facteur d'incrémention devient :

$$I_{n,i} = g_{n,i} - (g + I_{n,1} + \dots + I_{n,i-1} + I_{n-1,i+1} + \dots + I_{n-1,m})$$

Par suite la procédure générale de cet algorithme est :

i) Soit $g_n = P(g_{n-1} - I_{n \bmod m} / C_{n \bmod m})$ et on fait une mise à jour de $I_{n \bmod m}$ en le prenant égal à $g_n - (g_{n-1} - I_{n \bmod m})$.

ii) Remplacer n par $n+1$ et retourner en i).

THEOREME 1 [8] :

Les vecteurs $g_{n,i}$ convergent vers la solution du problème (4.1)-(5.2), à savoir g^k quand $n \rightarrow \infty$ pour $i=1,2,\dots,m$.

1.5. APPLICATIONS DE LA METHODE DE DYKSTRA :

1.5.1. Régression isotone :

Ecrivons les contraintes définissant le problème de *régression isotone* (P) sous la forme équivalente :

$$x_i - x_{i+1} \leq 0 \quad i=1,2,\dots,n-1.$$

Et posant

$$C_i = \{x \in \mathbb{R}^n : x_i - x_{i+1} \leq 0\},$$

Le cône des contraintes s'écrit alors sous la forme (5.2). De plus la projection de g sur C_i est explicite. Elle est donnée par :

$$P(g/C_i) = \begin{cases} g & \text{si } g_i - g_{i+1} \leq 0 \\ (g_1, g_2, \dots, g_{i-1}, \frac{g_i w_i + g_{i+1} w_{i+1}}{w_i + w_{i+1}}, \frac{g_i w_i + g_{i+1} w_{i+1}}{w_i + w_{i+1}}, g_{i+2}, \dots, g_n) & \text{si } g_i - g_{i+1} > 0. \end{cases}$$

1.5.2. REGRESSION CONCAVE :

Le problème de la régression concave est une extension naturelle du problème précédent, il a fait l'objet de plusieurs publications (HILDRETH 1954, HANSON and PLEDGER 1976, WU[19], R.DYKSTRA [8] et M.A. BOUGHAZI et PHAM DINH TAO[4]).

On se donne n valeurs (abscisses) $y_i, i=1, \dots, n$, telles que $y_1 < y_2 < \dots < y_n$. Le problème consiste à minimiser

$$\sum_{i=1}^n w_i (g_i - x_i)^2$$

sous les contraintes de concavités

$$\frac{x_{i+2} - x_{i+1}}{y_{i+2} - y_{i+1}} \leq \frac{x_{i+1} - x_i}{y_{i+1} - y_i} ; i=1,2,\dots,n-2.$$

que l'on écrit sous la forme (5.2) en posant, pour $i=1,2,\dots,n-2$:

$$C_i = \{x \in \mathbb{R}^n : \frac{x_{i+2} - x_{i+1}}{y_{i+2} - y_{i+1}} \leq \frac{x_{i+1} - x_i}{y_{i+1} - y_i}\}.$$

C_i est un cône fermé de \mathbb{R}^n . On peut donc calculer la projection de g sur le cône $C_i, i=1,2,\dots,n-2$, par l'expression :

$$[P_{C_i}(x)]_j = \begin{cases} x_i^* + b_i^*(y_j - y_i^*) & j = i, i+1, i+2 \\ x_j & j \neq i, i+1, i+2 \end{cases}$$

Où

$$x_i^* = \frac{\sum_{j=i}^{i+2} x_j w_j}{\sum_{j=i}^{i+2} w_j}$$

$$y_i^* = \frac{\sum_{j=i}^{i+2} y_j w_j}{\sum_{j=i}^{i+2} w_j}$$

et

$$b_i^* = \frac{\sum_{j=i}^{i+2} (x_j - x_i^*) y_j w_j}{\sum_{j=i}^{i+2} (y_j - y_i^*)^2 w_j}$$

5.4.3) Régression isotone pour un ordre général :

On considère le problème général de régression qui englobe comme cas particuliers les deux problèmes précédents. Il consiste à minimiser

$$\sum_{i=1}^n w_i (g_i - x_i)^2$$

sous les contraintes

$$\sum_{j=1}^n a_{ij} x_j \leq 0 \quad \text{pour } i=1, \dots, m.$$

Si on pose

$$C_i = \{x \in \mathbb{R}^n ; \sum_{j=1}^n a_{ij} x_j \leq 0\} \quad \text{pour } i=1, \dots, m.$$

Le problème est du même type que (5.1)-(5.2). De plus l'expression de la projection de g sur C_i est simple et explicite :

$$P(g/C_i) = \begin{cases} g & \text{si } \sum_{j=1}^n a_{ij} x_j \leq 0 \\ (g_1', g_2', \dots, g_n') & \text{si } \sum_{j=1}^n a_{ij} x_j > 0. \end{cases}$$

où

$$g_j' = g_j - \frac{\sum_{k=1}^n g_k a_{ik} a_{ij} w_j^{-1}}{\sum_{k=1}^n a_{ik}^2 w_k^{-1}} \quad \text{pour } j=1, \dots, n.$$

REFERENCES

- [1] **R.E.BARLOW, J.BARTHOLOMEW, M.BERMNER, and H.D.BRUNK(1972)**
Statistical Inference under Order Restrictions.
Wiley, New York.
- [2] **R.E.BARLOW and H.D.BRUNK(1972)**
The Isotonic Problem and its Dual.
J. Amer. Stat. Assoc, vol 67, No 337, pp. 140-147.
- [3] **M.A.BOUGHAZI, G.DROUET D'AUBIGNY and PHAM DINH TAO(1986)**
A Primal Dual Algorithm for Isotonic and Concave Regression Problem.
Submitted to Computational Statistics & Data Analysis.
- [4] **M.A.BOUGHAZI , G.DROUET D'AUBIGNY et PHAM DINH TAO(1987)**
Un algorithme primal dual pour le calcul du minimum d'une fonction convexe sur l'intersection de convexes et ses applications en analyse des données.
Rapp.Rech , N° 638M,TIM3 /IMAG, Grenoble.
- [5] **E.V.BURG and J.DE LEEUW(1983)**
Non Linear Canonical Correlation.
British J.of Math. and Statist. Psych. 36, pp.54-80.
- [6] **J.R.CHOI(1976)**
The Law of Sweep-the Negative in the Estimation under Order Restrictions.
Memoirs of the faculty of Sciences, Kyushu University, series A, Mathematics, vol 30, No1, pp. 135-143.
- [7] **R.DYKSTRA(1981)**
An Isotonic Regression Algorithm.
J.of Statist.Plann.and Inference 5, pp 355-363.
- [8] **R.DYKSTRA(1983)**
An Algorithm for Restricted Least Squares Regression.
J.Amer.Stat.Assoc, vol 78, No 384, pp.837-842.
- [9] **M.L.EATON(1984)**
Invariant Orderings, and Order Preservation, in Y.L.TONG (Ed.) Inequalities in Statistics and Propability(IMS Lecture Notes, vol. 5, pp.13-25.
- [10] **C.V.EEDEN(1956)**
Maximum Likelihood Estimation of Ordered Probabilities.
Proc.K.ned.Akad.Wet.(A), 59/Indag.Math.,18, pp.444-455.

[11]C.L.LAWSON and R.J.HANSON(1974)
Solving Least Squares Problems.
Prentice Hall, N.J.

[12]J.B.KRUSKAL(1964)
Multidimensional Scaling by Optimizing Goodness of Fit to a nonmetric Hypothesis. *Psychometrika* 29, pp 1-27 and 115-129.

[13]J.B.KRUSKAL(1965)
Analysis of Factorial Experiments by Estimating Monotone Transformation of the Data. *The J.of the Royal Statist.Soc.*, 27, pp.251-263.

[14]R.T.ROCKAFELAR(1970)
Convex Analysis.
Princeton University Press, Princeton.

[15]R.N.SHEPARD(1962)
The Analysis of Proximities : Multidimensional Scaling with an Unknown Distance Function, *Psychometrika* 27, pp 125-140 and 219-246.

[16]W.A.THOMPSON(1962)
The Problem of Negative Estimates of Variance Components.
Ann.Math.Statist. 33, pp 273-289.

[17]F.W.YOUNG, J.DE LEEUW and Y.TAKANE(1976)
Regression with Qualitative and Quantitative Variables: an Alternating Least Squares Method with Optimal Scaling Features, *Psychometrika* 41, pp 505-529.

[18]F.W.YOUNG, J.DE LEEUW and Y.TAKANE(1978)
Principal Components of Mixed Measurement Level Multivariate Data : an Alternating Least Squares Method with Optimal Scaling Features, *Psychometrika* 43, pp 279-281.

[19]Chien-Fu WU(1982)
Some Algorithms for Concave and Isotonic Regression.
TIMS Studies in Management Sciences, 19, pp. 105-116.

CHAPITRE II

***ETUDE ET ADAPTATION DE LA METHODE DE L'INVERSE PARTIEL A
LA RESOLUTION DES PROBLEMES DE REGRESSION.***



II.1. INTRODUCTION :

On va étudier, dans ce chapitre, le problème numérique posé par la minimisation d'une fonction convexe sur une intersection de convexes fermés en se ramenant à la recherche d'un zéro d'une somme (finie) d'opérateurs monotones maximaux, qu'on résoud la méthode de l'inverse partiel (J.E.SPINGARN[21]).

Dans le cas particulier où $f(\cdot)$ est une fonction quadratique définie positive, la minimisation de $f(\cdot)$ revient à construire le projecteur sur l'intersection de convexes fermés. Pour ce faire, on développe *une méthode nouvelle* qui s'inscrit dans le cadre des méthodes connues sous le nom de "*méthodes de projection*". Elle s'avère très simple à mettre en œuvre lorsque les projections sur chacun des convexes sont obtenues facilement.

Lorsque la matrice de produit scalaire qui définit la fonction objectif n'est pas diagonale, il est recommandé de transformer le premier problème, via les conditions de KUHN & TUCKER, en un *problème monotone complémentaire* dont le traitement présente la même difficulté que la résolution d'un système linéaire à matrice définie positive.

L'utilisation de la théorie de dualité de FENCHEL (R.T.ROCKAFELLAR[17]) nous permet de montrer que les algorithmes présentés, donnent non seulement la solution primale mais aussi celle du problème dual associé.

Il nous a de plus paru instructif d'explicitier la correspondance liant *les multiplicateurs de Lagrange* aux *variables duales* introduites dans la méthode de résolution proposée.

Ce chapitre s'achève par l'application de cette approche à la résolution de deux problèmes statistiques la régression isotone et la régression concave.

II.2. RAPPEL SUR LES OPERATEURS MAXIMAUX MONOTONES :

Avant d'exposer notre méthode de résolution du problème en termes de programmation mathématique, nous rappelons les concepts essentiels à la présentation de la méthode proposée, les opérateurs maximaux monotones, l'algorithme du point approximant, et l'opérateur inverse partiel. Pour plus de détails le lecteur se référera aux travaux de R.T.ROCKAFELLAR[17], [18], [19], J.J.MOREAU[12], J.E.SPINGARN[21], [22], [23].

a) Une multi-application $T: \mathbb{R}^n \rightarrow \mathbb{R}^n$ est dite monotone si :

$$\langle x-x', y-y' \rangle \geq 0 \text{ pour } y \in T(x), y' \in T(x').$$

Le graphe de T est l'ensemble $\text{Gr}(T) = \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^n ; y \in T(x)\}$. Si le graphe de T n'est contenu dans aucun autre graphe d'un opérateur monotone, T est dit maximal.

THEOREME 1 [20]: On note $\text{Dom}T = \{x \in \mathbb{R}^n ; T(x) \neq \emptyset\}$ le domaine effectif de T , et $\text{Int}C$ étant l'intérieur de l'ensemble C . Soient T_1 et T_2 deux opérateurs maximaux monotones. Si $\text{Int}(\text{Dom}T_1) \cap \text{Dom}T_2 \neq \emptyset$ (ou $\text{Dom}T_1 \cap \text{Int}(\text{Dom}T_2) \neq \emptyset$), alors $T = T_1 + T_2$ est maximal monotone.

Si D est un ensemble convexe fermé, on note $N_D(x)$ le cône normal à D en x défini par :

$$N_D(x) = \{y \in \mathbb{R}^n ; \langle y, x' - x \rangle \leq 0 \quad \forall x' \in D\} \text{ si } x \in D \text{ et } \emptyset \text{ sinon}$$

COROLLAIRE 2 [20]: Soient une multi-application maximale monotone $T_0: \mathbb{R}^n \rightarrow \mathbb{R}^n$ et D un sous ensemble convexe fermé sur \mathbb{R}^n , si l'une des conditions suivantes est satisfaite:

a) $\text{Dom}T_0 \cap \text{Int}D \neq \emptyset$; ou

b) $D \cap (\text{Int}(\text{Dom}T_0)) \neq \emptyset$;

alors la multi-application $T(x) = \{T_0(x) + N_D(x) \text{ si } x \in D \text{ et } \emptyset \text{ sinon}\}$ est maximale monotone.

Les opérateurs maximaux monotones possèdent de plus la propriété suivante MINTY[11] :

pour tout $x \in \mathbb{R}^n$, il existe un unique $P(x) \in \mathbb{R}^n$ tel que $x - P(x) \in T(P(x))$, qui permet de définir l'application proximante univoque de T

$$P: \mathbb{R}^n \rightarrow \mathbb{R}^n$$

Le problème fondamental posé par la recherche des zéros d'une fonction s'écrit dans le cas des opérateurs maximaux monotones :

$$\text{" Trouver } x \in \mathbb{R}^n \text{ tel que } 0 \in T(x) \text{ " (2.1) .}$$

L'application $P = \text{Prox}_T = (I+T)^{-1}$ qui a été introduite par J.J.MOREAU[12] permet de résoudre théoriquement un tel problème.

La méthode de Prox pour la résolution du problème (2.1) se fonde sur la propriété suivante :

$$x = \text{Prox}_T(x) = (I+T)^{-1}(x) \text{ si et seulement si } 0 \in T(x).$$

Pour un x^0 arbitraire cet algorithme génère une suite (x^k) de \mathbb{R}^n selon le processus itératif suivant :

$$x^{k+1} = \text{Prox}_T(x^k) = (I+T)^{-1}(x^k) \quad \text{pour } k=0,1,2,\dots$$

La convergence de la méthode des approximations successives permettant de calculer un point fixe de l'application Prox_T associée à un opérateur maximal monotone quelconque T a été étudiée très en détail par R.T.ROCKAFELLAR[18] qui a montré, sous réserve de l'existence d'un zéro de cet opérateur, que la suite engendrée converge toujours globalement vers un point fixe.

THEOREME 2 [18] : *La suite (x^k) définie ci-dessus converge vers un zéro de T , s'il en existe un. Si T n'admet pas de zéro, alors $\|x^k\|^2 \rightarrow +\infty$.*

b) Soient A, B deux sous-espaces complémentaires orthogonaux de \mathbb{R}^n (i.e., $A^\perp = B$ et $B^\perp = A$).

Tout élément $z \in \mathbb{R}^n$ s'écrit d'une manière unique comme $z = z_A + z_B$ avec z_A (resp. z_B) la projection de z sur A (resp. sur B).

SPINGARN[21] a introduit la notion d'inverse partiel d'un opérateur T relativement au sous-espace A , noté $T_A: \mathbb{R}^n \rightarrow \mathbb{R}^n$ dont le graphe est :

$$\text{Gr}(T_A) = \{(x_A + y_B, y_A + x_B) : y \in T(x)\}.$$

C'est à dire $v \in T_A(w)$ si et seulement si $w = x_A + y_B$ et $v = y_A + x_B$.

L'appellation d'inverse partiel est dû au fait que lorsque $A=\{0\}$, on récupère l'inverse ordinaire (i.e., $T_A=T^{-1}$), et lorsque $A=\mathbb{R}^n$, on obtient l'opérateur lui-même (i.e., $T_A=T$).

PROPOSITION 1 (SPINGARN[21]) : T_A est monotone (resp maximal) si et seulement si T est monotone (resp maximal).

J.E.SPINGARN[22] en introduisant le concept d'inverse partiel a étendu très largement le champ d'applications de la méthode Prox et permis ainsi de résoudre des problèmes qui ne pouvaient se traiter directement.

Ce concept a également conduit à de nouvelles et intéressantes méthodes de décomposition des problèmes [23]. Une classe importante de problèmes se formule ainsi :

$$\text{"Trouver } x \in A \text{ et } y \in B \text{ tel que } y \in T(x)\text{" (2.2).}$$

D'après les travaux de J.E.SPINGARN[22], ce problème est équivalent à:

$$\text{"Trouver } x \in A \text{ et } y \in B \text{ tel que } 0 \in T_A(x+y)\text{"}$$

On est conduit, pour résoudre (2.2), à trouver un zéro z de T_A . Or $0 \in T_A(z)$ implique que $x=z_A$ et $y=z_B$ sont solutions du problème (2.2).

Par suite, la méthode de l'inverse partiel introduite par J.E.SPINGARN n'est autre que l'application de la méthode Prox à la résolution du problème:

$$\text{"Trouver } z \in \mathbb{R}^n \text{ tel que } 0 \in T_A(z)\text{" (2.3).}$$

II.2.1. PRESENTATION DE LA METHODE DE L'INVERSE PARTIEL :

Il s'agit de trouver un point fixe de l'application $\text{Prox}_{T_A}=(I+T_A)^{-1}$ et ensuite de projeter z sur A et sur B ce qui fournit une solution primale x (resp. une solution duale y).

La méthode des approximations successives permet alors de calculer de façon itérative un zéro de T_A .

La résolution du problème (2.3) par application de la méthode Prox utilise le processus itératif suivant :

Pour un $z^0 \in \mathbb{R}^n$ arbitraire, l'algorithme génère une suite (z_k) définie par :

$$z^{k+1}=(I+T_A)^{-1}(z^k) \quad (2.4)$$

Le point z^k , donné par la $k^{\text{ième}}$ itération, s'écrit $z^k = x^k + y^k$ avec $x^k \in A$ et $y^k \in B$.
La relation (2.4) est équivalente à :

$$z^k - z^{k+1} \in T_A(z^{k+1}).$$

Or par définition de T_A , il existe x^k, y^k éléments de \mathbb{R}^n avec $y^k \in T(x^k)$ tels que

$$z^k - z^{k+1} \in T_A(z^{k+1}) \Leftrightarrow z^{k+1} = (x^k)_A + (y^k)_B \text{ et } z^k - z^{k+1} = (y^k)_A + (x^k)_B$$

D'où $z^k = z^{k+1} + (y^k)_A + (x^k)_B = [(x^k)_A + (y^k)_B] + [(y^k)_A + (x^k)_B] = x^k + y^k$.

Donc trouver z^{k+1} par la formule (2.4) équivaut à trouver x^k et y^k éléments de \mathbb{R}^n tels que

$$z^k = x^k + y^k = x^k + y^k \text{ et } y^k \in T(x^k)$$

puis prendre

$$x^{k+1} = (z^{k+1})_A = (x^k)_A \text{ et } y^{k+1} = (z^{k+1})_B = (y^k)_B.$$

Ainsi, l'algorithme Prox relativement à l'opérateur T_A , appliqué à la résolution du problème (2.2), prend la forme suivante :

Algorithme 1 (ou **méthode de l'inverse partiel**) :

étape 1 : $x^0 \in A, y^0 \in B$ choisis arbitrairement.

étape 2 : trouver $x^k \in \mathbb{R}^n, y^k \in \mathbb{R}^n$ tels que : $x^k + y^k = x^k + y^k$ et $y^k \in T(x^k)$.

étape 3 : $x^{k+1} = (x^k)_A$ et $y^{k+1} = (y^k)_B$.

retour à l'**étape 2** avec ces nouvelles valeurs .

Le théorème de convergence de cet algorithme (voir [21]) est le suivant :

THEOREME 1 : Soient (x^k) et (y^k) deux suites générées par l'algorithme 1. On a soit

- i) $x^k \rightarrow x^*$ et $y^k \rightarrow y^*$ faiblement, pour x^* et y^* solutions de (2.2); soit
- ii) $\|x^k + y^k\|^2 \rightarrow \infty$ et (2.2) n'admet pas de solution .

II.3. MINIMISATION D'UNE FONCTION CONVEXE SUR L'INTERSECTION DE CONVEXES FERMÉS :

On se propose de minimiser une fonction convexe $f(\cdot)$ sur l'intersection de convexes fermés. Considérons le problème (3.1) :

$$\min\{f(x) \ ; \ x \in \bigcap_{i=1}^m C_i\} \quad (P_2).$$

Ici $x \in \mathbb{R}^n$, et $f(\cdot)$ est une fonction convexe. Les $C_i \subset \mathbb{R}^n$ pour $i=1, \dots, m$ sont des ensembles convexes fermés.

Posons $C = \bigcap_{i=1}^m C_i$.

II.3.1. REDUCTION DU PROBLEME (3.1) A LA RECHERCHE D'UN ZERO D'UNE SOMME FINIE D'OPERATEURS MAXIMAUX MONOTONES :

Appelons χ_{C_i} la fonction indicatrice du convexe C_i :

$$\chi_{C_i}(x) = \begin{cases} 0 & \text{si } x \in C_i \\ +\infty & \text{sinon.} \end{cases}$$

Les fonctions $f, \chi_{C_i} \ i=1, \dots, m$ sont des fonctions convexes, propres et semi-continues inférieurement sur \mathbb{R}^n . Le problème (3.1) se traduit alors par le problème de minimisation sans contraintes sur \mathbb{R}^n suivant :

$$\min\{f(x) + \chi_C(x) \ ; \ x \in \mathbb{R}^n\} \quad (3.1).$$

Où χ_C est la fonction indicatrice de $C = \bigcap_{i=1}^m C_i$.

Il est facile de voir que :

$$\chi_C(x) = \chi_{\bigcap_{i=1}^m C_i}(x) = \sum_{i=1}^m \chi_{C_i}(x).$$

Le problème (3.1) devient alors :

$$\min\{F(x) \ ; \ x \in \mathbb{R}^n\} \quad (3.2).$$

Où $F(x) = f(x) + \sum_{i=1}^m \chi_{C_i}(x)$.

Le théorème qui suit donne une caractérisation de la solution de (3.1) :

THEOREME 1 [17]: Si f est une fonction convexe, propre et semi-continue inférieurement sur \mathbb{R}^n , les propriétés suivantes sont équivalentes:

- (a) f possède un minimum global au point x ;
- (b) f possède un minimum local au point x ;
- (c) $0 \in \partial f(x)$.

Selon ce **THEOREME**, Une condition nécessaire et suffisante pour que x^* soit solution du problème (3.2) est :

$$0 \in \partial F(x^*) \quad (3.3).$$

THEOREME 2 : Soient f_1, f_2 des fonctions convexes semi-continues inférieurement sur \mathbb{R}^n . Supposons qu'il existe $x_0 \in \mathbb{R}^n$ en lequel f_1 et f_2 sont finies, l'une d'entre elles étant continues en ce point. Alors on a :

$$\partial(f_1+f_2)(x) = \partial f_1(x) + \partial f_2(x)$$

Preuve : voir H.BREZIS[4], P.J.LAURENT[8] et R.T.ROCKAFELLAR[20].

COROLLAIRE 1 : Soient f_1, f_2, \dots, f_m des fonctions convexes semi-continues inférieurement sur \mathbb{R}^n . Supposons qu'il existe $x_0 \in \mathbb{R}^n$ en lequel toutes les f_i sont finies, et f_1, \dots, f_{m-1} sont continues en ce point. Alors on a :

$$\partial[f_1+f_2+\dots+f_m](x) = \partial f_1(x) + \partial f_2(x) + \dots + \partial f_m(x), \text{ pour tout } x \in \mathbb{R}^n$$

Preuve : voir P.J.LAURENT[8].

Selon (3.3) et le **COROLLAIRE 1** de cette partie, la minimisation de $F(x)$ sans contraintes équivaut à

$$0 \in \partial f(x^*) + \partial \chi_{C_1}(x^*) + \partial \chi_{C_2}(x^*) + \dots + \partial \chi_{C_m}(x^*) \quad (3.4)$$

D'autre par, si C est un convexe fermé on a :

$$\partial \chi_C(x^*) = N_C(x^*)$$

Où $N_C(x^*)$ est le cône normal à C en x^* qui se définit par :

$$N_C(x^*) = \{z \in \mathbb{R}^n \mid \langle z, x^* - x \rangle < 0 \quad \forall x \in C\}$$

La relation (3.4) devient alors ::

$$0 \in \partial f(x^*) + N_{C_1}(x^*) + N_{C_2}(x^*) + \dots + N_{C_m}(x^*) \quad (3.5).$$

PROPOSITION 1 : *Si f est propre, convexe et semi-continue inférieurement sur \mathbb{R}^n , alors ∂f (le sous différentiel de f) est une multi-application maximale monotone.*

Ce résultat a été établi par J.J.MOREAU[12] dans le cadre des espaces de Hilbert, alors que G.MINTY[11] avait déjà traité certains cas particuliers. R.T.ROCKAFELLAR[20] l'a ensuite étendu aux espaces de Banach.

∂f (resp $\partial \chi_{C_i} = N_{C_i}$ pour $i=1 \dots m$) est le sous différentiel d'une fonction convexe propre sur \mathbb{R}^n semi-continue inférieurement f (resp χ_{C_i} pour $i=1 \dots m$), par conséquent ∂f (resp χ_{C_i} , $i=1, \dots, m$) est un opérateur maximal monotone.

On est donc conduit à :

trouver un zéro d'une somme d'opérateurs maximaux monotones.

$$\text{"Trouver } x^* \in \mathbb{R}^n \text{ tel que } 0 \in \partial f(x^*) + N_{C_1}(x^*) + N_{C_2}(x^*) + \dots + N_{C_m}(x^*) \text{"} \quad (3.6).$$

Lemme 1 : *$x^* \in \mathbb{R}^n$ est solution du problème (3.1) si et seulement s'il existe $y_0 \in \partial f(x^*)$ et $y_i \in N_{C_i}(x^*)$ pour $i=1, \dots, m$ avec $y_0 + y_1 + \dots + y_m = 0$.*

Preuve :

En effet; si x^* est solution du problème (3.1), alors $0 \in \partial f(x^*) + N_{C_1}(x^*) + N_{C_2}(x^*) + \dots + N_{C_m}(x^*)$ d'où l'existence des y_i pour $i=0, 1, \dots, m$ tels que $y_0 \in \partial f(x^*)$ et $y_i \in N_{C_i}(x^*)$ pour $i=1, \dots, m$ avec $y_0 + y_1 + \dots + y_m = 0$.

Réciproquement, s'il existe $y_0 \in \partial f(x^*)$ et $y_i \in N_{C_i}(x^*)$ pour $i=1, \dots, m$ tels que $y_0 + y_1 + \dots + y_m = 0$, alors $0 \in \partial f(x^*) + N_{C_1}(x^*) + \dots + N_{C_m}(x^*)$, de plus $x^* \in C_1 \cap C_2 \cap \dots \cap C_m$.

On en déduit alors (THEOREME 1 du paragraphe II.3.1.) que x^* est solution du problème (3.6).

L'introduction des variables y_i nous permet de reconstituer le problème (3.6) sous la forme du problème (2.2).

Le **Lemme 1** du paragraphe II.3.1 permet d'écrire le problème (3.6) sous la forme (2.2). Pour ce faire, on définit :

$$\begin{aligned} \mathfrak{H} &= \mathbb{R}^n \times \mathbb{R}^n \times \dots \times \mathbb{R}^n \quad (m+1 \text{ fois}) \\ T &= \partial f \times N_{C_1} \times N_{C_2} \times \dots \times N_{C_m}, \\ A &= \{(x_0, x_1, \dots, x_m) \in \mathfrak{H} : x_0 = \dots = x_m\} \\ B &= \{(y_0, y_1, \dots, y_m) \in \mathfrak{H} : y_0 + \dots + y_m = 0\} \end{aligned}$$

On constate que $(y_0, \dots, y_m) \in T(x_0, \dots, x_m)$ si et seulement si $y_0 \in \partial f(x_0)$ et $y_i \in N_{C_i}(x_i)$; $i=1, \dots, m$.

$T: \mathfrak{H} \rightarrow \mathfrak{H}$ est maximal monotone, A et B sont deux sous-espaces complémentaires orthogonaux de \mathfrak{H} ($A=B^\perp$ et $B=A^\perp$).

Dans l'espace produit le problème (3.6) est équivalent au problème (3.7) :

Trouver $(x_0, \dots, x_m) \in A, (y_0, \dots, y_m) \in B$ tels que $(y_0, \dots, y_m) \in T(x_0, \dots, x_m)$

Si $(x_0^*, x_2^*, \dots, x_m^*) \in A$ et $(y_0^*, y_2^*, \dots, y_m^*) \in B$ sont solutions du problème (3.7), alors $x_0^* = x_2^* = \dots = x_m^*$ par définition de A, qu'on note $x^* \in C = \bigcap C_i$, et aussi $y_0^* \in \partial f(x^*), y_i^* \in N_{C_i}(x^*)$ pour $i=1, \dots, m$ avec $y_0^* + y_2^* + \dots + y_m^* = 0$. D'après le **lemme 1** du paragraphe II.3.1 x^* est alors solution du problème (3.6).

L'algorithme 1 de résolution du problème (3.7) décrit dans le paragraphe précédent se traduit ici par :

Initialisation : Soient $x^0 = (x^0, \dots, x^0)$ et $y^0 = (y_0^0, \dots, y_m^0)$ tels que $y_0^0 + \dots + y_m^0 = 0$.

Etape principale : Pour avoir $x^{k+1} \in A$ et $y^{k+1} \in B$ ($z^{k+1} = x^{k+1} + y^{k+1}$), soit $x^k \in A$ et $y^k \in B$ les points construits à l'étape k, calculons x^k et y^k tels que

$$x^k + y^k = x^k + y^k \text{ et } y^k \in T(x^k)$$

Ce qui équivaut à

$$x^k + y^{k,i} = x^{k,i} + y^{k,i} \text{ et } y^{k,i} \in T_i(x^{k,i}) \quad i=0, 1, \dots, m$$

Ou encore en remplaçant T_i par son expression :

$$x^{k+1}y^{k,i} = x^{k,i} + y^{k,i} \quad \text{et} \quad y^{k,i} \in \partial f(x^{k,i}) \quad \text{et} \quad y^{k,i} \in N_{C_i}(x^{k,i}) \quad i=1, \dots, m.$$

$x^{k+1} \in A$ et $y^{k+1} \in B$ sont alors donnés par

$$x^{k+1} = (x^k)_A = (x^+, x^+, \dots, x^+)$$

$$y_i^{k+1} = (y^{k,i})_B = (y_0^+, y_1^+, \dots, y_m^+)$$

Or $x^k = (x^+, x^+, \dots, x^+) + (x_0^+, x_1^+, \dots, x_m^+)$ avec $\sum_{i=0}^m x_i^+ = 0$.

D'autre part $x^k = (x^{k,0}, \dots, x^{k,m})$, d'où

$$x^+ + x_i^+ = x^{k,i} \quad \text{pour} \quad i=0, \dots, m \quad (3.8).$$

Sommons sur i les deux membres de l'égalité (3.8)

$$\sum_{i=0}^m (x^+ + x_i^+) = \sum_{i=0}^m x^{k,i} \Leftrightarrow (m+1)x^+ = \sum_{i=0}^m x^{k,i} \Leftrightarrow x^+ = \frac{1}{m+1} \sum_{i=0}^m x^{k,i}.$$

De même $y^k = (y^{k,0}, \dots, y^{k,m})$ s'écrit

$$y^k = (y^+, y^+, \dots, y^+) + (y_0^+, y_1^+, \dots, y_m^+) \quad \text{avec} \quad \sum_{i=0}^m y_i^+ = 0.$$

et donc pour $i=0, \dots, m$ on a $y^+ + y_i^+ = y^{k,i}$

$$\text{Soit} \quad y_i^+ = y^{k,i} - y^+ = y^{k,i} - \frac{1}{m+1} \sum_{i=0}^m y^{k,i} \quad \text{pour} \quad i=0, 1, \dots, m.$$

En résumé; l'algorithme 2 de résolution du problème (3.1) se décrit comme suit :

Algorithme 2 :

étape 1 : Soient $x \in \mathbb{R}^n$, $y_i \in \mathbb{R}^n$ pour $i=0, 1, \dots, m$ tels que $y_0 + y_1 + \dots + y_m = 0$

étape 2 : construire pour $i=0, \dots, m$ $x_i', y_i' \in \mathbb{R}^n$ tels que

$$x + y_i = x_i' + y_i' \quad \text{pour} \quad i=0, \dots, m$$

$$y_0' \in \partial f(x_0')$$

$$y_i' \in N_{C_i}(x_i') \quad \text{pour} \quad i=1, \dots, m$$

étape 3 :

$$x^+ = \frac{1}{m+1} \sum_{i=0}^m x_i' ,$$

$$\text{et } y_i^+ = y_i' - \frac{1}{m+1} \sum_{i=0}^m y_i' \text{ pour } i=0,1,\dots,m.$$

Retour à l'**étape 2** avec ces nouvelles valeurs.

En explicitant l'**étape 2** de cet algorithme on aura :

$$1) x+y_0=x_0'+y_0' \text{ et } y_0' \in \partial f(x_0') \quad (7.9)$$

et

$$2) x+y_i=x_i'+y_i' \text{ et } y_i' \in N_{C_i}(x_i') \text{ pour } i=1,\dots,m \quad (7.10)$$

Les relations (7.9) et (7.10) impliquent respectivement

$$x_0' \in (I+\partial f)^{-1}(x+y_0) \text{ et } x_i' \in (I+N_{C_i})^{-1}(x+y_i) \text{ pour } i=1,\dots,m.$$

La proximante $(I+N_{C_i})^{-1}$ de χ_{C_i} est l'opérateur de projection sur le convexe fermé C_i car pour tout $z \in C_i$:

$$y = (I+N_{C_i})^{-1}(x) \Leftrightarrow x-y \in N_{C_i}(y) \Leftrightarrow \langle x-y, z-y \rangle \leq 0 \Leftrightarrow y = \text{proj}_{C_i}(x),$$

d'où

$$x_i' = P_{C_i}(x+y_i) \quad i=1,\dots,m.$$

L'algorithme 2 devient alors :

étape 1 : on prend x, y_0, y_1, \dots, y_m éléments de \mathbb{R}^n tels que $y_0 + y_1 + \dots + y_m = 0$.

étape 2 : trouver x_i', y_i' pour $i=0, \dots, m$ éléments de \mathbb{R}^n tels que

$$x+y_i=x_i'+y_i' , \quad x_0' \in (I+\partial f)^{-1}(x+y_0) \text{ et}$$

$$x_i' = P_{C_i}(x+y_i) \quad i=1,\dots,m.$$

étape 3 :

$$x^+ = \frac{1}{m+1} \left[x_0' + \sum_{i=0}^m P_{C_i}(x+y_i) \right] ,$$

$$\text{et } y_i^+ = y_i' - \frac{1}{m+1} \sum_{i=0}^m y_i' \text{ pour } i=0,1,\dots,m.$$

Retour à l'étape 2 avec ces nouvelles valeurs.

Remarque : La seule opération compliquée dans cet algorithme est le calcul de l'application Prox de l'opérateur sous différentiel ∂f à savoir le calcul de $(I+df)^{-1}$.

Cependant, ALART[1] dans sa thèse propose justement une méthode pour résoudre un tel problème et de ce fait l'algorithme ainsi décrit serait implémentable.

II.4. MINIMISATION D'UNE FONCTION QUADRATIQUE DEFINIE POSITIVE SUR L'INTERSECTION DE CONES CONVEXES FERMES :

On s'intéresse dans cette partie à la minimisation d'une forme quadratique définie positive sur une intersection de cônes convexes fermés. Ce problème est un cas particulier de celui étudié au paragraphe précédent. En le résolvant par la méthode décrite ci-dessus cela revient à le décomposer en sous-problèmes ne mettant chacun en jeu qu'un seul des cônes, et du fait que les cônes qu'on considère sont des demi-espaces, la méthode développée ici serait facile à mettre en œuvre.

II.4.1. POSITION DU PROBLEME ET HYPOTHESE :

Soit \mathbb{R}^n muni du produit scalaire usuel et de la norme euclidienne associée notés (\cdot, \cdot) et $\|\cdot\|$. Soient $C_i \subset \mathbb{R}^n$; pour $i=1, \dots, m$ des cônes convexes fermés définis ainsi :

$$C_i = \{x \in \mathbb{R}^n ; \langle a_i, x \rangle \leq 0\} \quad i=1, \dots, m, \quad \text{avec } a_i \text{ un élément donné de } \mathbb{R}^n.$$

$$C = \bigcap_{i=1}^m C_i.$$

Et le problème :

$$\text{"Trouver } x^* \in C \text{ tel que } f(x^*) = \min\{f(x) ; x \in C\} \text{", (4.1)}$$

avec $f(x) = 1/2 \langle x, Hx \rangle + \langle x, d \rangle$ où $d \in \mathbb{R}^n$.

Nous ferons les hypothèses suivantes :

(H₁) : La matrice H est symétrique.

(H₂) : La forme quadratique associée à la matrice est définie positive :

$$x \neq 0 \Rightarrow \langle x, Hx \rangle > 0$$

(H₃) : Le minimum absolu n'appartient pas à C :

$$x = -H^{-1}d \notin C.$$

(H₄) : $C \neq \emptyset$

On suppose que le problème (4.1) admet un point $x^* \in C$ tel que :

$$f(x^*) = \min\{f(x) ; x \in C\}.$$

En se basant sur l'étude faite au paragraphe II.3, on décrit l'algorithme pour obtenir la solution du problème (4.1).

II.4.2. DESCRIPTION DE L'ALGORITHME .:

L'algorithme général décrit précédemment peut être particularisé pour donner une méthode de résolution du problème (4.1). En effet, dans ce cas la fonction $f(\cdot)$ est donnée par : $f(x) = 1/2 \langle x, Hx \rangle + \langle x, d \rangle$ ainsi que les convexes $C_i, i=1, \dots, m$.

De ce fait, l'algorithme pour la résolution de (4.1) prend la forme suivante :

étape 1 : on prend x, y_0, y_1, \dots, y_m éléments de \mathbb{R}^n tel que $y_0 + y_1 + \dots + y_m = 0$.

étape 2 : trouver x'_i, y'_i pour $i=0, \dots, m$ éléments de \mathbb{R}^n tels que

$$x + y_i = x'_i + y'_i, ,$$

$$(I+H)x'_0 = (x + y_0) - d \text{ et}$$

$$x'_i = P_{C_i}(x + y_i) \quad i=1, \dots, m.$$

étape 3 :

$$x^+ = \frac{1}{m+1} \left[x'_0 + \sum_{i=0}^m P_{C_i}(x + y_i) \right] ,$$

$$\text{et } y_i^+ = y'_i - \frac{1}{m+1} \sum_{i=0}^m y'_i \text{ pour } i=0, 1, \dots, m.$$

retour à l'étape 2 avec ces nouvelles valeurs.

Remarque 1: Cet algorithme donne en plus de la solution primale x^* , les solutions "duales" y_i pour $i=0, \dots, m$. Nous montrons dans le paragraphe II.6. que la variable y_0 est solution du problème dual associé. On établit au paragraphe II.7. un lien entre les autres variables $y_i, i=1, \dots, m$ et les multiplicateurs de Lagrange.

Remarque 2 : L'exécution de cet algorithme nécessite la résolution, à chaque étape, d'un système linéaire à matrice définie positive ainsi que le calcul des projections sur chacun des cônes considérés.

II.4.3. REDUCTION DU PROBLEME DE PROGRAMMATION QUADRATIQUE EN UN PROBLEME MONOTONE COMPLEMENTAIRE :

Dans cette partie, on va présenter une deuxième méthode de résolution du problème (4.1) exprimée dans le contexte des opérateurs maximaux monotones. Pour ce faire, nous montrons que les conditions de KUHN & TUCKER permettent de réduire le problème de programmation quadratique en un *problème monotone complémentaire* bien connu en programmation mathématiques.

Cette approche est principalement justifiée surtout dans le cas où la matrice qui définit la fonction quadratique n'est pas diagonale. Il est en effet préférable de résoudre le problème dual du problème posé. Soit donc à minimiser la fonction :

$$f(x) = 1/2 \langle x, Hx \rangle + \langle x, d \rangle \quad (4.2)$$

$$\text{s.t} \quad Ax \leq b \quad (4.3)$$

où H est une matrice symétrique définie positive d'ordre n, $A = [a_1, a_2, \dots, a_m]^t$ est une matrice $m \times n$ et d est un vecteur de \mathbb{R}^n . Un *problème monotone complémentaire* (4.4) s'exprime sous la forme :

" Trouver $x \in \mathbb{R}^n, y \in \mathbb{R}^m$ tels que

$$y \in T(x), \quad x \geq 0, \quad y \geq 0 \quad \text{et} \quad \langle x, y \rangle = 0 "$$

Où $T : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est un opérateur monotone maximal .

PROPOSITION 1 : Le problème (4.2)-(4.3) est équivalent à :

" Trouver μ, λ éléments de \mathbb{R}^m tels que

$$\mu = AH^{-1}A^t\lambda + AH^{-1}d, \quad \mu \geq 0, \lambda \geq 0 \quad \text{et} \quad \langle \mu, \lambda \rangle = 0 "$$

Preuve:

On note par λ le vecteur multiplicateur de Lagrange associé aux contraintes $Ax \leq b$ et on pose $\mu = -Ax + b$. Les conditions de KUHN & TUCKER du problème (4.2)-(4.3) s'écrivent :

$$Hx + d + A^t\lambda = 0$$

$$\langle Ax + b, \lambda \rangle = 0.$$

et donc puisque H est symétrique définie positive

$$x = -H^{-1}A^t\lambda - H^{-1}d \quad (4.5)$$

D'où

$$\mu = -Ax + b = AH^{-1}A^t\lambda + AH^{-1}d + b.$$

Nous définissons ainsi une fonction affine et donc monotone maximal :

$$T(\lambda) = AH^{-1}A^t\lambda + AH^{-1}d + b$$

Le problème ainsi transformé s'écrit :

(4.6) " Trouver $\mu, \lambda \in \mathbb{R}^m$ tel que

$$\mu = AH^{-1}A^t\lambda + AH^{-1}d + b, \quad \mu \geq 0, \lambda \geq 0 \text{ et } \langle \mu, \lambda \rangle = 0 "$$

Lemme 1 : Si les vecteurs $a_i, i=1, \dots, m$ sont linéairement indépendants la matrice AA^t est régulière.

Preuve :

Soit $y \in \mathbb{R}^n$ un vecteur non nul tel que :

$$AA^ty = 0 \quad (4.7)$$

alors

$$y^tAA^ty = (A^ty)^t(A^ty) = \|A^ty\|^2 = 0 \Leftrightarrow A^ty = 0$$

Mais $A^ty = y_1a_1 + y_2a_2 + \dots + y_ma_m$ c'est à dire une combinaison linéaire des vecteurs a_i , à coefficients y_i pour $i=1, \dots, m$, où les y_i sont les composantes du vecteur y .

Comme les a_i sont par hypothèse linéairement indépendants alors A^ty ne peut pas être nulle sans que y ne soit pas nul, donc (4.7) est impossible. Ainsi la matrice AA^t n'annule qu'un vecteur nul, d'où sa régularité.

PROPOSITION 2 : Si (μ^*, λ^*) est solution du problème (4.6) alors $(-H^{-1}A^t\lambda^* - H^{-1}d)$ est solution du problème (4.2)-(4.3). Et si x^* est solution du problème (4.2)-(4.3) alors $(-Ax^* + b, -(AH^{-1}A^t)^{-1}Ax^* - (AH^{-1}A^t)^{-1}AH^{-1}d)$ est solution de (4.6).

Preuve :

D'après les conditions de KUHN & TUCKER on a $Hx^* + d + A^t\lambda^* = 0$ d'où $x^* = -H^{-1}A^t\lambda^* - H^{-1}d$ est solution du problème (4.2)-(4.3).

or on sait que $\mu^* = -Ax^* + b$, d'où

$\mu^* = -Ax^* + b = AH^{-1}d + AH^{-1}A'\lambda^* + b$, on aura donc

$$Ax^* + AH^{-1}d + AH^{-1}A'\lambda^* = 0$$

d'où $\lambda^* = -(AH^{-1}A')^{-1}Ax^* - (AH^{-1}A')^{-1}H^{-1}d$,

d'où

$$(\mu^*, \lambda^*) = (-Ax^* + b, -(AH^{-1}A')^{-1}Ax^* - (AH^{-1}A')^{-1}H^{-1}d)$$

est solution du problème (4.6).

Soit maintenant χ_{\geq} la fonction indicatrice de l'orthant positif défini ainsi

$$\chi_{\geq}(\lambda) = \begin{cases} 0 & \text{si } \lambda \geq 0 \\ \infty & \text{si } \lambda < 0 \end{cases}$$

PROPOSITION 3 : Le sous différentiel de χ_{\geq} est l'ensemble

$$S(\lambda) = \partial\chi_{\geq}(\lambda) = \begin{cases} \{\mu ; \mu \leq 0, \langle \lambda, \mu \rangle = 0\} & \text{si } \lambda \geq 0 \\ \emptyset & \text{si } \lambda < 0 \end{cases}$$

Preuve :

Par définition du sous différentiel de χ_{\geq} en un point x on a

$$\partial\chi_{\geq}(\lambda) = \{\mu \in \mathbb{R}^m ; \chi_{\geq}(\lambda') - \chi_{\geq}(\lambda) \geq \langle \mu, \lambda' - \lambda \rangle \quad \forall \lambda' \in \mathbb{R}^m\}$$

on a $\partial\chi_{\geq}(\lambda) = \{\mu \in \mathbb{R}^m ; \langle \mu, \lambda' - \lambda \rangle \leq 0 \quad \forall \lambda' \geq 0\}$ si $\lambda \geq 0$

en particulier pour $\lambda' = 2\lambda$ et $\lambda' = 0$ on a $\langle \mu, \lambda \rangle \leq 0$ et $\langle \mu, \lambda \rangle \geq 0$, et donc pour $\lambda \geq 0$

$$\partial\chi_{\geq}(\lambda) = \{\mu \in \mathbb{R}^m ; \langle \mu, \lambda \rangle \leq 0\}$$

Si $\lambda < 0$ alors $\chi_{\geq}(\lambda) = \infty$ d'où $\partial\chi_{\geq}(\lambda) = \emptyset$.

C'est à dire $\partial\chi_{\geq}(\lambda) = S(\lambda)$. Ainsi S est le sous différentiel de la fonction indicatrice χ_{\geq} et donc, d'après la PROPOSITION 1 paragraphe II.3.1, maximal monotone.

$(I+S)^{-1}$ est la matrice de l'opérateur de la projection sur l'orthant positif car

$$\mu=(I+S)^{-1}(\lambda) \Leftrightarrow \lambda-\mu \in S(\mu) \Leftrightarrow \langle \lambda-\mu, \mu \rangle=0 .$$

C'est à dire $\mu=\pi(\lambda)$ où π définit la projection sur l'orthant positif ce qui équivaut à dire que $\pi(\lambda)$ représente la partie positive de λ .

Le problème (4.6) est équivalent alors à:

Trouver $\lambda \in \mathbb{R}^n$ tel que

$$0 \in T(\lambda)+\partial \chi_{\geq}(\lambda)=T(\lambda)+S(\lambda).$$

En effet;

$$\begin{aligned} 0 \in T(\lambda)+S(\lambda) &\Leftrightarrow -T(\lambda) \in S(\lambda) \Leftrightarrow -T(\lambda) \leq 0, \langle -T(\lambda), \lambda \rangle=0 \text{ et } \lambda \geq 0 \\ &\Leftrightarrow T(\lambda) \geq 0, \langle T(\lambda), \lambda \rangle=0 \text{ et } \lambda \geq 0 \end{aligned}$$

Ce problème est un cas particulier du problème (3.6) traité au paragraphe II.3.1. La méthode de résolution du problème (4.6) se déduit directement de l'algorithme 1 en prenant $T_1=T$ et S avec

$$A=\{(\lambda_1, \lambda_2) \in \mathbb{R}^m \times \mathbb{R}^m ; \lambda_1=\lambda_2\} \text{ et}$$

$$B=\{(\mu_1, \mu_2) \in \mathbb{R}^m \times \mathbb{R}^m ; \mu_1+\mu_2=0\}.$$

Algorithme 3 :

étape 1: Soient λ, μ_1, μ_2 arbitraires dans \mathbb{R}^m tels que $\mu_1+\mu_2=0$

étape 2 : Trouver $\lambda'_1, \lambda'_2, \mu'_1$ et μ'_2 tels que

$$\mu_1+\lambda=\lambda'_1+\mu'_1 \quad \mu'_1 \in T(\lambda'_1) \text{ et}$$

$$\mu_2+\lambda=\lambda'_2+\mu'_2 \quad \mu'_2 \in T(\lambda'_2)$$

étape 3 :

$$\lambda^+ = \frac{1}{2}\lambda'_1 + \frac{1}{2}\lambda'_2$$

$$\mu_1^+ = \frac{1}{2}\mu'_1 - \frac{1}{2}\mu'_2$$

$$\mu_2' = \frac{1}{2}\mu_2' - \frac{1}{2}\mu_1'$$

Retour à l'étape 2 avec ces nouvelles valeurs.

L'étape 2 de l'algorithme s'explique ainsi :

$$\mu_1 + \lambda = \lambda_1' + \mu_1' \text{ et } \mu_1' \in T(\lambda_1') \Leftrightarrow (I+T)\lambda_1' = \mu_1 + \lambda \text{ et } \mu_1' = \mu_1 + \lambda - \lambda_1'$$

$$\mu_2 + \lambda = \lambda_2' + \mu_2' \text{ et } \mu_2' \in S(\lambda_2') \Leftrightarrow (I+S)\lambda_2' = \mu_2 + \lambda \text{ et } \mu_2' = \mu_2 + \lambda - \lambda_2' \Leftrightarrow$$

$$\Leftrightarrow \lambda_2' = (I+S)^{-1}(\mu_2 + \lambda) = \pi(\mu_2 + \lambda) \text{ et } \mu_2' = \mu_2 + \lambda - \pi(\mu_2 + \lambda) = -\pi(-\mu_2 - \lambda)$$

Compte-tenu du fait que $\lambda_2' = \pi(\mu_2 + \lambda)$, $\mu_2 = -\mu_1$, $\mu_2' = -\pi(-\mu_2 - \lambda)$ et $T_1(\lambda) = AH^{-1}A'\lambda + AH^{-1}d$, et en posant $\mu = m_1$ l'algorithme 3 s'écrit alors ainsi :

étape 1 : μ, λ éléments de $(\mathbb{R}^m)_+$ choisis arbitrairement.

étape 2 : trouver μ', λ' éléments de $(\mathbb{R}^m)_+$ tels que :

$$(I+AH^{-1}A')\lambda' = \mu + \lambda - AH^{-1}d \text{ et}$$

$$\mu + \lambda = \mu' + \lambda'$$

étape 3 :

$$\lambda = \frac{1}{2}\lambda + \frac{1}{2}\pi(\lambda - \mu).$$

$$\mu = \frac{1}{2}\mu' + \frac{1}{2}\pi(\mu - \lambda).$$

Retour à l'étape 2 avec ces nouvelles valeurs.

On remarque que cet algorithme conduit à la résolution à chaque itération d'un système linéaire à matrice symétrique définie positive et à calculer la projection sur l'orthant positif.

Une fois λ^* est donnée par l'algorithme, x^* solution du problème (4.2)-(4.3) se déduit directement par :

$$x^* = H^{-1}A\lambda^* - H^{-1}d.$$

II.5. UNE METHODE NOUVELLE DE CONSTRUCTION DE LA "PROJECTION SUR L'INTERSECTION DE CONVEXES FERMES ":

Le problème de la construction du projecteur de l'intersection de convexes fermés a fait l'objet de plusieurs travaux, on cite entre autres ceux de R.DYKSTRA[6] , HAUGAZEAU[7], G.PIERRA[13] , et P.J.LAURENT et MARTINET[9] . Les algorithmes présentés dans ([6], [7], [13]) reposent sur la possibilité d'effectuer facilement les projections sur chacun des convexes initiaux. L'idée de base de ces algorithmes est de n'utiliser à chaque itération qu'un seul convexe.

On développe dans cette partie une méthode nouvelle qui s'inscrit dans la classe des méthodes de projections. Elle diffère des autres par le fait qu'à chaque itération on met en jeu tous les convexes dans un premier temps puis on calcule leurs barycentres en un deuxième.

II.5.1. POSITION DU PROBLEME ET HYPOTHESES :

Soit l'espace \mathbb{R}^n muni du produit scalaire usuel et de sa norme euclidienne associée.

$C_i \subset \mathbb{R}^m$ pour $i=1, \dots, m$ des convexes fermés.

$$C = \bigcap_{i=1}^m C_i$$

$g \in \mathbb{R}^m$ élément donné. On se propose de résoudre le problème suivant :

$$\text{"trouver } x^* \in \mathbb{R}^n \text{ tel que } 1/2 \|g - x^*\|^2 = \min\{1/2 \|g - x\|^2 ; x \in C\} \text{" (5.1).}$$

Nous supposons que le problème n'est pas dégénéré et supposons de plus vérifiées les hypothèses.

$$(H_1) \quad C = \bigcap_{i=1}^m C_i \neq \emptyset.$$

$$(H_2) \quad g \notin C.$$

Le problème (5.1) consiste à chercher la projection de g sur l'intersection de convexes fermés et nous proposons d'obtenir cette projection en utilisant des projections simultanées sur les convexes $C_i, i=1 \dots m$.

Le problème (5.1) est un cas particulier du problème général (4.1), ainsi exploitons nous l'idée du paragraphe précédent, pour résoudre (5.1) .

Il est donc naturel de proposer de résoudre (5.1) à l'aide de l'algorithme 3, en remplaçant dans celui-ci la matrice H par la matrice identité I, ce qui nous permet de donner l'algorithme 4.

Algorithme 4 :

étape 1: $x^0 \in \mathbb{R}^n, y^{0,i} \in \mathbb{R}^n$ pour $i=0, \dots, m$ tels que $y^{0,0} + y^{0,1} + \dots + y^{0,m} = 0$.

étape 2 : trouver $x^{k,i}, y^{k,i} \in \mathbb{R}^n$ pour $i=0, \dots, m$ tels que

$$x^{k,0} = (x^k + y^{k,0} + g) / 2 \quad \text{et} \quad x^k + y^{k,0} = x^{k,0} + y^{k,0}.$$

$$x^{k,i} = P_{C_i}(x^k + y^{k,i}) \quad i=1, \dots, m \quad \text{et} \quad x^k + y^{k,i} = x^{k,i} + y^{k,i} \quad i=1, \dots, m.$$

étape 3 :

$$x^{k+1} = \frac{1}{m+1} \left[\frac{x^k + y^{k,0} + g}{2} + \sum_{i=1}^m P_{C_i}(x^k + y^{k,i}) \right].$$

$$y^{k+1,i} = y^{k,i} - \frac{1}{m+1} \sum_{i=0}^m y^{k,i}, \quad \text{pour } i=0, 1, \dots, m.$$

Retour à l'étape 2 avec ces nouvelles valeurs .

THEOREME 1 [22] : Soient (x^k) et $(y^{k,i})$ pour $i=1, \dots, m+1$ des suites générées par l'algorithme 4 On a :

$x^k \rightarrow x^* \in \cap C_i, y^{k,i} \rightarrow y_0^* \in \partial f(x^*)$ et $y^{k,i} \rightarrow y_i^* \in N_{C_i}(x^*)$ pour $i=1, \dots, m$ tels que $y_0^* + y_1^* + \dots + y_m^* = 0$.

Cet algorithme est une méthode nouvelle de construction du projecteur sur l'intersection des convexes fermés C_i pour $i=1, \dots, m$, qui s'ajoute à la famille des méthodes bien connues sous le nom de "méthodes de projection", s'avère facile à mettre en œuvre lorsque la projection sur chacun des convexes initiaux est aisée à obtenir .

Remarque 1 : L'algorithme 3 donne en plus de la solution primale x^* les solutions "duales" $y_0^*, y_1^*, \dots, y_m^*$. Nous montrons dans le paragraphe qui suit que la variable duale y_0^* est solution du problème dual.

Remarque 2 : L'algorithme 3 peut être implémenté sur calculateur parallèle.

II.6. DUALITE DE FENCHEL POUR LE PROBLEME (4.1) :

Dans cette partie, on montre que l'*algorithme 2* donne en plus de la solution primale la solution du problème dual associé. Soit le problème :

$$\min \left\{ \frac{1}{2} \langle x, Hx \rangle + \langle x, d \rangle ; x \in \bigcap_{i=1}^m C_i \right\} \quad (6.1).$$

Où les C_i sont définis par : $C_i = \{x \in \mathbb{R}^n ; \langle a_i, x \rangle \leq 0\}$ pour $i=1, \dots, m$ avec $a_i \in \mathbb{R}^n$.

Le problème (6.1) est équivalent à :

$$\min \{ 1/2 \langle x, Hx \rangle + \langle x, d \rangle + \chi_C(x) ; x \in \mathbb{R}^n \} \quad (6.1).$$

En posons $f(x) = 1/2 \langle x, Hx \rangle + \langle x, d \rangle$ et $h(x) = -\chi_C(x)$

Le problème (6.1) s'écrira :

$$\inf \{ f(x) - h(x) ; x \in \mathbb{R}^n \} \quad (6.2)$$

Le problème dual de (6.2) est :

$$\sup \{ h^*(y) - f^*(y) ; y \in \mathbb{R}^n \}, \quad (6.3)$$

Où f^* (resp $h^* = (-\chi_C)^*$) est la fonction conjuguée de f (resp de $h = -\chi_C$) définie ainsi :

$$f^*(y) = \sup \{ \langle x, y \rangle - f(x) ; x \in \mathbb{R}^n \}$$

$$(\chi_C)^*(y) = \sup \{ \langle x, y \rangle + \chi_C(x) ; x \in \mathbb{R}^n \} = \sup \{ \langle x, y \rangle ; x \in C \} = \chi_{C^*}(y)$$

La fonction $(\chi_C)^*$ est appelée *fonction support* de C (cf Rockafellar[19]).

$$\chi_{C^*}(y) = \begin{cases} 0 & \text{si } y \in C^* \\ +\infty & \text{sinon} \end{cases}$$

En faisant les calculs on aura :

$$f^*(y) = 1/2 \langle y, H^{-1}y \rangle - \langle y, H^{-1}d \rangle$$

$$h^*(y) = (-\chi_C)^*(y) = -\chi_{C^*}(-y)$$

THEOREME DE FENCHEL 1 [17] : Soient f une fonction convexe propre sur \mathbb{R}^n , et soit h une fonction concave propre sur \mathbb{R}^n . Alors si $ri(\text{Dom } f) \cap ri(\text{Dom } h) \neq \emptyset$

$$\inf\{f(x) - h(x) ; x \in \mathbb{R}^n\} = \sup\{h^*(x) - f^*(x) ; x \in \mathbb{R}^n\}$$

(et notons $ri(\text{dom } f)$ l'intérieur relatif du domaine de f).

Le problème (6.3) s'écrira alors :

$$\sup\{-\chi_{C^*}(-y) - [1/2\langle y, H^{-1}y \rangle - \langle y, H^{-1}d \rangle] ; y \in \mathbb{R}^n\}.$$

D'où le problème dual (6.3):

$$-\inf\{1/2\langle y, H^{-1}y \rangle - \langle y, H^{-1}d \rangle ; -y \in C^*\} \quad (6.4).$$

La relation de polarité entre l'intersection et la somme des convexes fermés:

$$C^* = (C_1 \cap \dots \cap C_m)^* = C_1^* + \dots + C_m^*.$$

donne enfin

$$-\inf\{1/2\langle y, H^{-1}y \rangle - \langle y, H^{-1}d \rangle ; -y \in C_1^* + \dots + C_m^*\} \quad (Q_5).$$

THEOREME DE FENCHEL 2 [17] : Soit f une fonction convexe propre semi-continue inférieurement sur \mathbb{R}^n . Soit C un cône convexe fermé dans \mathbb{R}^n et C^* son cône polaire.

i) Si $ri(\text{Dom } f) \cap ri(C) \neq \emptyset$, on a

$$f(x^*) = \inf\{f(x) ; x \in C\} = - \inf\{f^*(y) ; -y \in C^*\} = -f^*(-y^*) \quad (6.6).$$

ii) En général, $(x^*, y^*) \in C \times C^*$ réalise le minimum de (6.6) si et seulement si :

- (1) $y^* \in \partial f(x^*)$
- (2) $x^* \in C$
- (3) $-y^* \in C^*$
- (4) $\langle x^*, y^* \rangle = 0$.

PROPOSITION: 1 :

La variable y_0^* donnée par l'algorithme 2 est solution du problème dual (6.5).

Preuve : D'après le THEOREME 2 de dualité de W.FENCHEL x^*, y^* satisfont

$$f(x^*) = \inf\{f(x) ; x \in C\} = -\inf\{f^*(y) ; -y \in C^*\} = -f^*(-y^*)$$

si et seulement si

$$y^* \in \partial f(x^*), x^* \in C, -y^* \in C^* \text{ et } \langle x^*, y^* \rangle = 0.$$

il suffit donc de vérifier, si $x^* \in C$ est solution du problème primal, que

$$y_0^* \in \partial f(x^*), -y_0^* \in C_1^* + \dots + C_m^*, \langle x^*, y_0^* \rangle = 0.$$

a) Montrons que $y_0^* \in \partial f(x^*)$: D'après le THEOREME 1 du paragraphe II.5.1 de convergence de l'algorithme 3 on a :

$$y_0^k \rightarrow y_0^* \in \partial f(x^*) \text{ où } x^* \text{ est solution du problème (5.1).}$$

b) Montrons que $-y_0^* \in C_1^* + \dots + C_m^*$: D'après le théorème de convergence 2 on a :

$$x^k \rightarrow x^* \in C_1 \cap \dots \cap C_m$$

$$y_0^k \rightarrow y_0^* \in \partial f(x^*)$$

$$\text{et } y_i^k \rightarrow y_i^* \in N_{C_i}(x^*) ; i=1, \dots, m \text{ avec } y_0^* + y_1^* + \dots + y_m^* = 0.$$

Montrons tout d'abord que $y_i^* \in C_i$ pour $i=1, \dots, m$.

$$y_i^* \in N_{C_i}(x^*) \Leftrightarrow \langle y_i^*, x - x^* \rangle \leq 0 \text{ pour tout } x \in C_i \quad (6.7)$$

Les C_i sont des cônes convexes et $x^* \in C_i$ pour $i=1, \dots, m$ alors $2x^* \in C_i$ pour $i=1, \dots, m$. Substituons la valeur $x=2x^*$ dans la relation(6.7), on aura :

$$\langle y_i^*, x^* \rangle \leq 0 \text{ pour } i=1, \dots, m \quad (6.8)$$

La relation (6.7) s'écrit alors :

$$\langle y_i^*, x \rangle \leq \langle y_i^*, x^* \rangle \quad \forall x \in C_i \text{ pour } i=1, \dots, m$$

En tenant compte de la relation (6.8) on aura :

$$\langle y_i^*, x \rangle \leq 0 \quad \forall x \in C_i \text{ pour } i=1, \dots, m$$

Ce qui implique $y_i^* \in (C_i)^*$ pour $i=1, \dots, m$.

Donc, On a bien

$$y_1^* + y_2^* + \dots + y_m^* \in (C_1)^* + (C_2)^* + \dots + (C_m)^* .$$

Or, d'après le THEOREME 1 du paragraphe II.5. :

$$-y_0^* = y_1^* + y_2^* + \dots + y_m^* , \text{ par conséquent}$$

$$-y_0^* \in (C_1)^* + (C_2)^* + \dots + (C_m)^* .$$

c) Montrons que $\langle x^*, y_0^* \rangle = 0$:

D'après le THEOREME 1 du paragraphe II.5. on a :

$$y_i^* \in N_{C_i}(x^*) \text{ pour } i=1, \dots, m, \text{ d'où}$$

$$\langle y_i^*, x - x^* \rangle \leq 0 \quad \forall x \in C_i \text{ pour } i=1, \dots, m.$$

En particulier pour $x = 2x^*$ et $x = 0$, car les C_i sont des cônes convexes de la forme

$$C_i = \{x \in \mathbb{R}^n ; \langle a_i, x \rangle \leq 0 \text{ avec } a_i \in \mathbb{R}^n\} \text{ pour } i=1, \dots, m,$$

on obtient :

$$\langle y_i^*, x^* \rangle \geq 0 \text{ et } \langle y_i^*, -x^* \rangle \leq 0 \text{ ce qui implique } \langle y_i^*, x^* \rangle = 0 \text{ pour } i=1, \dots, m.$$

D'où $\langle y_1^* + y_2^* + \dots + y_m^*, x^* \rangle = 0$ or $-y_0^* = y_1^* + y_2^* + \dots + y_m^*$ ce qui donne $\langle y_0^*, x^* \rangle = 0$.

D'après a), b) et c) y_0^* est bien solution du problème dual (6.5).

II.7. CORRESPONDANCE ENTRE L'INVERSE PARTIEL ET LES MULTIPLICATEURS DE LAGRANGE :

Le problème qu'on veut résoudre est le suivant

$$\min \{f(x) ; x \in C_1 \cap C_2 \cap \dots \cap C_m\} \quad (7.1)$$

où les C_i sont définis par : $C_i = \{x \in \mathbb{R}^n ; f_i(x) \leq 0\}$ pour $i=1, \dots, m$

Avec les $f_i(\cdot)$ $i=1, \dots, m$ sont des fonctions convexes différentiables et à valeurs finies, Alors

$$\chi_C = \chi_{\cap C_i} = \chi_{C_1} + \dots + \chi_{C_m}$$

D'où;

$$\partial \chi_C(x) = \partial \chi_{C_1}(x) + \dots + \partial \chi_{C_m}(x)$$

Ce qui est équivalent à :

$$N_C(x) = N_{C_1}(x) + \dots + N_{C_m}(x)$$

On peut facilement vérifier que :

$$N_{C_i}(x) = \begin{cases} \{\lambda_i \nabla f_i(x) ; \lambda_i \geq 0\} & \text{si } f_i(x) = 0 \\ \{0\} & \text{si } f_i(x) < 0 \end{cases} \quad (7.2).$$

D'après le théorème de convergence de l'algorithme on a:

$$x^k \rightarrow x^* \in C \text{ solution du problème}$$

$$y_0^k \rightarrow y_0^* \in \partial f(x^*)$$

$$y_i^k \rightarrow y_i^* \in N_{C_i}(x^*) \quad i=1, \dots, m \text{ avec } y_0^* + y_1^* + \dots + y_m^* = 0$$

D'après la relation (7.2) il existe $\lambda_i^* \geq 0$ pour $i=1, \dots, m$ tels que :

$$y_i^* = \lambda_i^* \nabla f_i(x^*) \quad (7.3)$$

$$\lambda_i^* = 0 \text{ si } f_i(x^*) < 0 \quad (7.4)$$

$$\lambda_i^* \geq 0 \text{ si } f_i(x^*) = 0 \quad (7.5).$$

PROPOSITION 1 : Soient λ_i^* pour $i=1,2,\dots,m$ désignant les multiplicateurs de Lagrange associés aux contraintes du problème (7.1) et y_i^* les variables duales données par la méthode de l'inverse partiel, on a alors la relation liant les multiplicateurs λ_i^* aux variables y_i^* :

$$y_i^* = \lambda_i^* \nabla f_i(x^*) \text{ pour } i=1,2,\dots,m.$$

Preuve : D'après le théorème de KUHN & TUCKER, et sous l'hypothèse que les f_i pour $i=1,2,\dots,m$ sont différentiables et sous l'hypothèse de qualification de SLATER, alors une condition nécessaire et suffisante pour que x^* soit minimum de f sur l'ensemble C est qu'il existe des nombres réels (appelés *multiplicateurs de Lagrange*) $\lambda_i \geq 0$ pour $i=1,2,\dots,m$ tels que :

$$0 \in \nabla f(x^*) + \lambda_1 \nabla f_1(x^*) + \lambda_2 \nabla f_2(x^*) + \dots + \lambda_m \nabla f_m(x^*)$$

et

$$\lambda_i f_i(x^*) = 0 \text{ pour } i=1,2,\dots,m.$$

Pour montrer que les λ_i^* pour $i=1,\dots,m$ définis par $y_i^* = \lambda_i^* \nabla f_i(x^*)$ sont les multiplicateurs de Lagrange il suffit de vérifier que :

$$0 \in \nabla f(x^*) + \lambda_1^* \nabla f_1(x^*) + \lambda_2^* \nabla f_2(x^*) + \dots + \lambda_m^* \nabla f_m(x^*)$$

et

$$\lambda_i^* f_i(x^*) = 0 \text{ pour } i=1,2,\dots,m.$$

En effet, d'après le théorème de convergence on a

$$y_0^* + y_1^* + \dots + y_m^* = 0, \quad y_0^* \in \nabla f(x^*) \text{ et } y_i^* \in N_{C_i}(x^*).$$

Ce qui implique $y_0^* = -(y_1^* + \dots + y_m^*) \in \nabla f(x^*)$ d'où

$$0 \in \nabla f(x^*) + (y_1^* + \dots + y_m^*). \tag{7.6}$$

Compte de (7.3), la relation (7.6) s'écrit ainsi :

$$0 \in \nabla f(x^*) + \lambda_1^* \nabla f_1(x^*) + \lambda_2^* \nabla f_2(x^*) + \dots + \lambda_m^* \nabla f_m(x^*)$$

et

$$\lambda_i^* f_i(x^*) = 0 \text{ pour } i=1,2,\dots,m.$$

e sont bien les conditions de KUHN-TUCKER, donc les λ_i^* ainsi définis sont les multiplicateurs de Lagrange.

II.8. APPLICATIONS:

II.8.1. REGRESSION ISOTONE :

a) Formulation du problème :

Le problème de regression isotone au sens des moindres carrés est défini par :

$$\min\left\{\sum_{i=1}^n w_i(g_i - x_i)^2 : x_i - x_{i+1} \leq 0 \text{ pour } i=1, \dots, n-1\right\} \quad (8.1).$$

Après transformation (8.1) est équivalent

$$\min\{1/2 \langle x, Wx \rangle + \langle x, W(-g) \rangle; x \in \cap C_i\}, \quad (8.2).$$

où les convexes C_i sont définis par:

$$C_i = \{x \in \mathbb{R}^n; x_i - x_{i+1} \leq 0\} \text{ pour } i=1, \dots, n-1,$$

et W est une $n \times n$ -matrice diagonale à éléments positifs, donnée par :

$$W = \begin{vmatrix} w_1 & 0 & \dots & \dots & 0 \\ 0 & w_2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \dots & w_n \end{vmatrix}$$

Nous signalons que pour un x donné, la projection sur C_i , pour $i=1, \dots, n-1$, est explicite et facile à formuler :

$$P_{C_i}(x) = \begin{cases} x & \text{si } x \in C_i \\ x & \text{si } x \notin C_i \end{cases}$$

où

$$x = (x_1, \dots, \frac{w_i x_i + w_{i+1} x_{i+1}}{w_i + w_{i+1}}, \frac{w_i x_i + w_{i+1} x_{i+1}}{w_i + w_{i+1}}, x_{i+2}, \dots, x_n)$$

L'application de l'*algorithme 3* au problème (8.2) est totalement décrit en précisant l'étape 2 :

Trouver $x_i', y_i' \in \mathbb{R}^n$ $i=0, \dots, n-1$ tels que

$$- (I+W)x_0' = x+y_0 + Wg \text{ et } x+y_0 = x_0' + y_0'$$

$$- x_i' = P_{C_i}(x+y_i) \text{ } i=1, \dots, n-1 \text{ et } x+y_i = x_i' + y_i' \text{ } i=1, \dots, n-1.$$

En explicitant ces égalités on obtient :

$$x_0' = \left(\frac{x_0+y_{01}+w_1g_1}{1+w_1}, \dots, \frac{x_n+y_{0n}+w_n\dot{g}_n}{1+w_n} \right)$$

et

$$x_i' = \begin{cases} x_i+y_i & \text{si } (x_i+y_{ii}) - (x_{i+1}+y_{ii+1}) \leq 0 \\ (x_i')' & \text{sinon} \end{cases}$$

où

$$(x_i')' = (x_1+y_{i1}, \dots, \frac{w_i(x_i+y_{ii})+w_{i+1}(x_{i+1}+y_{ii+1})}{w_i+w_{i+1}}, \frac{w_i(x_i+y_{ii})+w_{i+1}(x_{i+1}+y_{ii+1})}{w_i+w_{i+1}}, x_{i+2}+y_{ii+2}, \dots, x_n+y_{in}).$$

b) Description de l'algorithme de résolution du problème de régression isotone :

Initialisation : $x^0, y^0, i \in \mathbb{R}^n$; $i=0, \dots, n-1$ tel que $y^{0,0} + y^{0,1} + \dots + y^{0,n-1} = 0$.

à l'itération $k+1$ on aura :

$$x^{k+1} = \frac{1}{n} \left[\frac{x^k + y^{k,0} + Wg}{2} + \sum_{i=1}^{n-1} P_{C_i}(x^k + y^{k,i}) \right]$$

$$y^{k+1,0} = \frac{x^k + y^{k,0} - Wg}{2} - \frac{1}{n} \left[\frac{x^k + y^{k,0} - Wg}{2} + \sum_{i=1}^{n-1} [x^k + y^{k,i} - P_{C_i}(x^k + y^{k,i})] \right].$$

$$y^{k+1,i} = x^k + y^{k,i} - P_{C_i}(x^k + y^{k,i}) - \frac{1}{n} \left[\frac{x^k + y^{k,0} - Wg}{2} + \sum_{i=1}^{n-1} [x^k + y^{k,i} - P_{C_i}(x^k + y^{k,i})] \right], \text{ pour } i=1, \dots, n-1.$$

L'algorithme ainsi décrit est *une méthode nouvelle de projections simultanées* pour la résolution du problème de la régression isotone qui s'avère très facile à mettre en œuvre. Elle diffère de celle de DYKSTRA[6], par le fait que ce dernier utilise des projections successives.

II.8.2. REGRESSION CONCAVE :

Le problème de régression concave est :

$$\min\left\{\sum_{i=1}^n w_i(g_i-x_i)^2 ; \frac{x_{i+2}-x_{i+1}}{y_{i+2}-y_{i+1}} \leq \frac{x_{i+1}-x_i}{y_{i+1}-y_i} : i=1,\dots,n-2\right\} \quad (8.3)$$

où $w_i > 0$ pour $i=1,\dots,n$ et $0 < y_1 < y_2 < \dots < y_n$.

Le problème peut être écrit d'une façon équivalente comme:

$$\min\{1/2\langle x, Wx \rangle + \langle x, W(-g) \rangle ; x \in C_1 \cap C_2 \cap \dots \cap C_{n-2}\} \quad (8.4)$$

où $C_i = \{x \in \mathbb{R}^n ; (y_{i+2} - y_{i+1})x_i - (y_{i+2} - y_i)x_{i+1} + (y_{i+1} - y_i)x_{i+2} \leq 0\}$ pour $i=1,\dots,n-2$.

La matrice W est la même que celle définie à la partie a) du paragraphe **II.8.1**.

La projection de x sur C_i est obtenue simplement puisque :

$$[P_{C_i}(x)]_j = \begin{cases} x_i^* + b_i^*(y_j - y_i^*) & j=i, i+1, i+2 \\ x_j & j \neq i, i+1, i+2 \end{cases}$$

Où

$$x_i^* = \frac{\sum_{j=i}^{i+2} x_j w_j}{\sum_{j=i}^{i+2} w_j}$$

$$y_i^* = \frac{\sum_{j=i}^{i+2} y_j w_j}{\sum_{j=i}^{i+2} w_j}$$

et

$$b_i^* = \frac{\sum_{j=i}^{i+2} (x_j - x_i^*) y_j w_j}{\sum_{j=i}^{i+2} (y_j - y_i^*)^2 w_j}$$

La résolution du problème (8.4) procède selon *l'algorithme 3*. On décrit le processus itératif analogue à celui présenté pour le problème précédent où seules les projections changent. Cette fois-ci on projette sur les C_i définis par les contraintes de concavité alors que dans (8.2), on projetait sur les C_i définis par les contraintes d'isotonie.

II.8.3. FORMULATION DU PROBLEME DE REGRESSION EN TERMES DE PROBLEME MONOTONE COMPLEMENTAIRE :

Dans cette partie, une deuxième formulation des problèmes de régression est présentée. Elle consiste à mettre ces problèmes sous forme de problèmes monotones complémentaires, problème déjà traité dans le paragraphe II.4.3.. Cette façon de faire permet de donner une autre manière de résolution de ces problèmes qui pourrait être parfois avantageuse et commode.

a) REGRESSION ISOTONE :

Reprenons le problème de régression isotone qui consiste à

$$\min \left\{ \sum_{i=1}^n w_i (g_i - x_i)^2 ; Ax \leq 0 \right\} \quad (8.5)$$

Où $w_i > 0$ (le poids), g un élément donné de \mathbb{R}^n et A une matrice d'ordre $(n-1) \times n$ définie ainsi :

$$A = \begin{bmatrix} 1 & -1 & 0 & \dots & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 \\ & & & \ddots & & \\ & & & & \ddots & \\ & & & & & \ddots & \\ & & & & & & -1 \\ 0 & 0 & \dots & \dots & \dots & \dots & 1 \end{bmatrix}$$

Le problème (8.5) est équivalent au problème suivant :

$$\min \left\{ 1/2 \langle x, Wx \rangle - \langle x, Wg \rangle ; Ax \leq 0 \right\} \quad (8.6)$$

Ici W est une matrice diagonale d'ordre $n \times n$ dont les éléments sont les poids w_i pour $i=1,2,\dots,n$.

D'après la PROPOSITION 1 du paragraphe II.4.3., (8.3) est équivalent au problème monotone complémentaire suivant :

Trouver $\mu, \lambda \in \mathbb{R}^{n-1}$ tels que

$$\begin{aligned} \mu &= AW^{-1}A'\lambda - Ag, \\ \mu &\geq 0, \\ \lambda &\geq 0 \text{ et} \\ \langle \mu, \lambda \rangle &= 0 \end{aligned}$$

L'algorithme pour la résolution de ce dernier problème consiste à partir d'un point (μ^0, λ^0) arbitraire de construire les suites (μ^k) , (λ^k) et (λ^k) $k=0,1,\dots$ de la manière suivante :

1) (μ^k, λ^k) donnent μ^{k+1} par la relation

$$(I+AW^{-1}A^t) \mu^{k+1} = AW^{-1}A^t (\mu^k + \lambda^k) - Ag$$

2) μ^{k+1} , μ^k et λ^k donnent μ^{k+1} et λ^{k+1} par la relation

$$\lambda^{k+1} = 1/2 \lambda^{k+1} + 1/2 \pi (\lambda^k - \mu^k)$$

$$\mu^{k+1} = 1/2 \mu^{k+1} + 1/2 \pi (\mu^k - \lambda^k)$$

La seule opération difficile à exécutée dans cet algorithme est la résolution du système linéaire à matrice symétrique définie positive :

$$(I+AW^{-1}A^t) \mu^{k+1} = AW^{-1}A^t (\mu^k + \lambda^k) - Ag$$

Où la matrice $AW^{-1}A^t$ est donnée par :

$$AW^{-1}A^t = \begin{bmatrix} \frac{1}{w_1} + \frac{1}{w_2} & -\frac{1}{w_2} & 0 & \dots & 0 \\ -\frac{1}{w_2} & \frac{1}{w_2} + \frac{1}{w_3} & -\frac{1}{w_3} & 0 & \dots & 0 \\ & & & & & -\frac{1}{w_{n-1}} \\ & & & & -\frac{1}{w_{n-1}} & \frac{1}{w_{n-1}} + \frac{1}{w_n} \end{bmatrix}$$

Ainsi, en procédant selon cette méthode, le problème se ramène à une répétition d'applications d'une procédure standard pour la résolution du système précédent.

Une fois la valeur de λ est calculée, on détermine la valeur de la solution du problème par la relation

$$x^* = -W^{-1}A^t \lambda + W^{-1}g$$

tout calcul fait la valeur explicite de x^* est égale à :

$$x^{*t} = \left(\frac{g_1 - \lambda_1}{w_1}, \frac{\lambda_1}{w_2} + \frac{g_2 - \lambda_2}{w_2}, \dots, \frac{\lambda_{n-2}}{w_{n-1}} + \frac{g_{n-1} - \lambda_{n-1}}{w_{n-1}}, \frac{g_n + \lambda_n}{w_n} \right).$$

II.8.3.2. REGRESSION CONCAVE :

Le problème de la régression concave est donnée par

$$\min \left\{ \sum_{i=1}^n w_i (g_i - x_i)^2 \ ; \ Ax \leq 0 \right\}, \tag{8.6}$$

où $w_i > 0$ (les poids) pour $i=1,2,\dots,n$, g un élément donné de \mathbb{R}^n et A une matrice d'ordre $n-1 \times n$ définie ainsi :

$$A = \begin{bmatrix} y_3 - y_2 & -(y_3 - y_1) & y_2 - y_1 & 0 & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & y_4 - y_3 & -(y_4 - y_2) & y_3 - y_2 & 0 & \dots & \dots & \dots & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & y_n - y_{n-1} & -(y_n - y_{n-2}) & y_{n-1} - y_{n-2} & \dots & \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix}$$

Les variables y_i vérifient la relation $0 < y_1 < y_2 < \dots < y_m$.

Le problème de régression concave se résoud de la même manière que celui de la régression isotone, sauf que dans ce cas-ci la matrice $AW^{-1}A^t$ serait symétrique définie positive pentadiagonale dont la forme générale de ses lignes est comme suit :

$$(AW^{-1}A^t)_1 = \left(\frac{(y_3 - y_2)^2}{w_1} + \frac{(y_3 - y_1)^2}{w_2} + \frac{(y_2 - y_1)^2}{w_3}, \frac{(y_3 - y_1)(y_4 - y_3)}{w_2}, \frac{(y_2 - y_1)(y_5 - y_4)}{w_3}, 0, \dots, 0 \right)$$

$$(AW^{-1}A^t)_2 = \left[\left(\frac{(y_3 - y_1)(y_4 - y_3)}{w_2} + \frac{(y_2 - y_1)(y_4 - y_2)}{w_3} + \frac{(y_4 - y_3)^2}{w_2} + \frac{(y_4 - y_2)^2}{w_3} + \frac{(y_3 - y_2)^2}{w_4}, \right. \right.$$

$$\left. \frac{(y_4 - y_2)(y_5 - y_4)}{w_3} + \frac{(y_3 - y_2)(y_5 - y_3)}{w_4}, \frac{(y_3 - y_2)(y_6 - y_5)}{w_4}, 0, \dots, 0 \right).$$

$$(AW^{-1}A^t)_i = \left[\begin{array}{l} (0,0,0,\dots, \frac{(y_{i-1}-y_{i-2})(y_{i+2}-y_{i+1})(y_{i+1}-y_{i-1})(y_{i+2}-y_{i+1})(y_i-y_{i-1})(y_{i+2}-y_i)}{w_i}, \frac{(y_{i+2}-y_{i+1})(y_{i+1}-y_{i-1})(y_{i+2}-y_{i+1})(y_i-y_{i-1})(y_{i+2}-y_i)}{w_i}, \frac{(y_{i+2}-y_{i+1})(y_{i+1}-y_{i-1})(y_{i+2}-y_{i+1})(y_i-y_{i-1})(y_{i+2}-y_i)}{w_{i+1}}, \\ \frac{(y_{i+2}-y_{i+1})^2}{w_i} + \frac{(y_{i+2}-y_i)^2}{w_{i+1}} + \frac{(y_{i+1}-y_i)^2}{w_{i+2}}, \\ \frac{(y_{i+2}-y_i)(y_{i+3}-y_{i+2})}{w_{i+1}} + \frac{(y_{i+1}-y_i)(y_{i+3}-y_{i+1})}{w_{i+2}}, \frac{(y_{i+1}-y_i)(y_{i+4}-y_{i+2})}{w_{i+2}}, 0,0,\dots,0) \end{array} \right]$$

et

$$(AW^{-1}A^t)_n = \left[\begin{array}{l} (0,0,\dots,0, \frac{(y_{n-3}-y_{n-4})(y_n-y_{n-1})(y_{n-1}-y_{n-3})(y_n-y_{n-1})(y_{n-2}-y_{n-3})(y_n-y_{n-2})}{w_{n-2}}, \frac{(y_{n-3}-y_{n-4})(y_n-y_{n-1})(y_{n-1}-y_{n-3})(y_n-y_{n-1})(y_{n-2}-y_{n-3})(y_n-y_{n-2})}{w_{n-2}}, \frac{(y_{n-3}-y_{n-4})(y_n-y_{n-1})(y_{n-1}-y_{n-3})(y_n-y_{n-1})(y_{n-2}-y_{n-3})(y_n-y_{n-2})}{w_{n-1}}, \\ \frac{(y_n-y_{n-1})^2}{w_{n-2}} + \frac{(y_n-y_{n-2})^2}{w_{n-1}} + \frac{(y_{n-1}-y_{n-2})^2}{w_n} \end{array} \right]$$

REFERENCES

- [1] **R.BARLOW, D.BARTHOLOMEW, J.BREMNER and H.BRUNK (1972)**
Statistical Inference Under Order Restriction.
John Wily & Son's, London.
- [2] **R.BARLOW and H.BRUNK (1972)**
The Isotonic Regression Problem and its Dual
J.of the American Statistical Association March 1972, vol 67, Nb 337
- [3] **M.BOUGHAZI, G.DROUET D'AUBIGNY and PHAM DINH TAO(1986)**
A Dual Algorithm for Isotonic and for Concave Regression
Congrés Compstat (Rome 1-5 sept 1986) Submitted to J.A.S.A.
- [4] **H.BREZIS (1973)**
Opérateurs maximaux monotones
North-Holland Mathematics Studies 5.
- [5] **W.CHIEN-FU(1983)**
Some Algorithms for Concave and Isotonic Regression
Studies in Manegement Sciences 19 p 105-116.
- [6] **R.DYKSTRA (1983)**
An Algorithm for Restricted Least Squares Regression
J.of the American Statistical Association Dec 1983, vol 78, Nb 384.
- [7] **HAUGAZEAU (1967)**
Sur les inéquations variationnelles et la minimisation de fonctionnelles convexes. Thèse, Paris, 7 juin 1967.
- [8] **P.J.LAURENT (1972)**
Approximation et optimisation.
Hermann, Paris.
- [9] **P.J.LAURENT et B.MARTINET (1967)**
Méthodes duales pour calcul du minimum d'une fonction convexe sur une intersection de convexes. Colloque d'Optimisation de Nice, juin 1967.
- [10] **B.MARTINET (1972)**
Détermination approchée d'un point fixe d'une application pseudo-contractante. Cas de l'application Prox.
C.R.Acad.Sc.Paris Sér A, 274: 163-165.
- [11] **G.MINTY (1962)**
Monotone (nonlinear) Operators in Hilbert Space.
Duke Math J. 29: 341-346.

[12] **J.J.MOREAU (1965)**

Proximité et dualité dans un espace Hilbertien.
Bull Soc Math France 93: 273-299.

[13] **G.PIERRA (1976)**

Méthode de décomposition et croisement d'algorithmes pour des problèmes d'optimisation. Thèse , Grenoble , 1976.

[14] **R.T.ROCKAFFELLAR (1968)**

A General Correspondence between Dual Minimax Problems and Convex Programs. *Pacific J Math* 25: 597-611.

[15] **R.T.ROCKAFFELLAR (1970)**

Monotone Operators Associated with Saddle Fonctions and Minimax problems. *American Math Society Providence* pp 241-250.

[16] **R.T.ROCKAFFELLAR (1970)**

On the Maximality of Sums of Nonlinear Monotone Operators.
Trans Amer Math Soc 149: 75-88.

[17] **R.T.ROCKAFFELLAR (1972)**

Convex Analysis.
Princeton University Press , Princeton.

[18] **R.T.ROCKAFFELLAR (1976)**

Monotone Operators and Proximal Point Algorithm.
S.I.A.M J Control Optim 14: 877-898.

[19] **R.T.ROCKAFFELLAR (1976)**

Augmented Lagrangians and the Proximal Point Algorithm in Convex Programming. *Math O.R* 1: 97-116.

[20] **R.T.ROCKAFFELLAR (1979)**

La théorie des sous-gradient et ses applications en optimisation.
Les presses de l'université de Montréal.

[21] **J.E.SPINGARN (1980)**

Fixed and Variable Constraints in Sensitivity Analysis.
SIAM Journal on Control and Optimisation 18: 297-310

[22] **J.E.SPINGARN (1983)**

Partial Inverse of Monotone Operators.
App.Math.Optim 10: 247-265.

[23] **J.E.SPINGARN (1985)**

Applications of the Method of Partial Inverses to Convex Programming : Decomposition. *Mathematical Programming* 32: 199-223.

CHAPITRE III

***ADAPTATION DE LA METHODE DE PIVOTAGE COMPLEMENTAIRE
A LA RESOLUTION DES PROBLEMES DE REGRESSION.***



IV.1. INTRODUCTION :

Ce chapitre montre comment les problèmes de programmation quadratique sous contraintes linéaires (voir [7]) peuvent se formuler en termes de programmation linéaire complémentaire moyennant les conditions de KUHN-TUCKER.

Dans un premier temps, nous exposons un ensemble de rappels sur la programmation linéaire ainsi que sa résolution par la méthode du simplexe découverte par G.DANTZIG en 1951. Celle-ci constitue en effet une composante de la méthode développée par C.LEMKE en 1969 pour résoudre un programme linéaire complémentaire.

Nous présentons ensuite le problème linéaire complémentaire noté par la suite PLC ainsi que l'algorithme de résolution par la *méthode de pivotage complémentaire* ou algorithme de C.LEMKE. Cette méthode a pour propriété essentielle de converger en un nombre fini d'itérations vers une solution du problème lorsque celle-ci existe.

Nous abordons alors les conditions de convergence de cet algorithme. Puis nous montrons comment on peut formuler un problème de programmation quadratique sous la forme PLC.

Nous exposons en fin de ce chapitre deux applications privilégiées, du fait de leur utilisation fréquente en Statistique et Analyse des Données, le problème de la *régression isotone* et celui de la *régression concave* aux sens des moindres carrés.

IV.2. PROGRAMMATION LINEAIRE :

IV.2.1. FORME STANDARD D'UN PROGRAMME LINEAIRE :

Le problème posé en programmation linéaire, consiste à optimiser une fonction linéaire sous contraintes linéaires. La formulation mathématique du programme s'écrit :

$$(P_1) \quad \begin{aligned} & \min\{z = \langle c, x \rangle ; x \in \mathbb{R}^n\} \\ & A \cdot x = b \\ & x \geq 0 \end{aligned}$$

Où A est une matrice réelle $m \times n$ (dite matrice des contraintes) de rang m , $c = (c_1, c_2, \dots, c_n)$ est le vecteur-ligne des coûts, $b = (b_1, b_2, \dots, b_m)^t$ le m -vecteur des seconds membres, et $z = \langle c, x \rangle = \sum c_i \cdot x_i$ est la fonction à minimiser ou fonction objectif.

IV.2.2. BASES, BASES REALISABLES, SOLUTIONS DE BASE :

DEFINITIONS ([1], [8]) :

Les colonnes de toute sous matrice carrée régulière B ($m \times m$) extraite de A (il en existe au moins une, puisque $\text{rang}(A) = m$), définissent une base de „ m ”. Par abus de langage on dit alors que B est une base.

Soit donc B une base, alors en permutant les colonnes de A , on peut toujours écrire A sous la forme $A = [B, N]$ où N est la sous matrice formées par les colonnes de A qui ne sont pas dans la base, et l'on peut de même partitionner x en $[x_B, x_N]^t$ et c en $[c_B, c_N]^t$.

Toute solution de (P_1) vérifie $A \cdot x = b$, c'est à dire :

$$B \cdot x_B + N \cdot x_N = b \quad (2.1).$$

On appelle solution de base (associée la base B), la solution particulière de (2.1) obtenue en imposant $x_N = 0$. x_B est alors déterminé de façon unique par la résolution du système linéaire :

$$B \cdot x_B = b \quad \text{soit} \quad x_B = B^{-1} \cdot b.$$

Une solution de base est dite réalisable si $x_B \geq 0$, autrement dit si :

$$B^{-1} \cdot b \geq 0.$$

Une base correspondant à une solution de base réalisable est appelée base réalisable.

une solution de base est dite dégénérée si une au moins des composantes du vecteur $x_B = B^{-1}.b$ est nulle.

IV.2.3. ALGORITHME DU SIMPLEXE ([1] , [8]) :

L'algorithme du simplexe découvert en 1951 par G.DANTZIG est le plus connu des algorithmes de résolution d'un programme linéaire. Sa célébrité est due à son efficacité constatée dans de nombreuses applications concrètes où il fût utilisé pour résoudre des programmes linéaires qui trouvaient leur origine dans des secteurs très variés de la vie économique. Notons d'ailleurs que cette efficacité pratique reste encore mal expliquée aujourd'hui.

L'algorithme du simplexe se décompose généralement en deux phases :

La phase (I) de l'algorithme consiste à vérifier si le système d'équations et d'inéquations définissant les contraintes de (P_1)

$$x \geq 0 ; A.x = b$$

admet ou non une solution réalisable. Dans le cas où le système n'a aucune solution réalisable, l'algorithme est terminé.

La phase (II) de l'algorithme du simplexe consiste à résoudre le programme linéaire (P_1) à partir de la solution de base réalisable fournie par la phase (I).

IV.2.4. FINITUDE DE L'ALGORITHME DU SIMPLEXE :

On dit que le programme linéaire est non dégénéré si à chaque itération les variables de base de la solution de base obtenue à l'entrée sont strictement positives. Ceci implique qu'à chaque nouvelle itération, le second membre b vérifie $b > 0$.

THEOREME 1 [8] : *Supposons que le programme linéaire reste non dégénéré à chaque itération du simplexe, alors, l'algorithme du simplexe converge en un nombre fini d'itérations.*

IV.3. LE PROBLEME LINEAIRE COMPLEMENTAIRE :

IV.3.1. POSITION DU PROBLEME PLC :

Le problème linéaire complémentaire consiste à Trouver $w \in \mathbb{R}^m$, $z \in \mathbb{R}^m$ tels que:

$$w - Mz = q \quad (1.1)$$

$$(P_2) \quad w_i \geq 0, z_i \geq 0 \text{ pour } i=1, \dots, m \quad (1.2)$$

$$w_i \cdot z_i = 0 \text{ pour } i=1, \dots, m \quad (1.3).$$

Où M est une matrice d'ordre $m \times m$, et q est un vecteur de \mathbb{R}^m .

REMARQUES :

- (w_i, z_i) forme un couple de composantes complémentaires.
- La composante w_i (resp z_i) est dite variable de base si $w_i \geq 0$ (resp $z_i \geq 0$).
- On dit que w_i (resp z_i) est une variable hors base si $w_i = 0$ (resp $z_i = 0$).

DEFINITION 1 :

Une solution (w^*, z^*) du système (P_2) est dite solution de base réalisable complémentaire si :

- 1) (w^*, z^*) est une solution de base réalisable de (1.1)-(1.2).
- 2) une composante du couple (w_i, z_i) est une variable de base pour $0 \leq i \leq m$.

La solution du problème (P_2) est triviale, si q est positif ou nul, et donnée par $w=q$ et $z=0$.

Lorsqu'il existe un i tel que $q_i < 0$, on introduit un nouveau vecteur colonne $\mathbf{1}$, dont toutes les composantes sont égales à 1, et une variable artificielle z_0 , pour définir un nouveau système :

$$w - Mz - \mathbf{1}z_0 = q \quad (1.4)$$

$$(P_3) \quad w_i \geq 0, z_i \geq 0, z_0 \geq 0 \text{ pour } i=1, \dots, m \quad (1.5)$$

$$w_i \cdot z_i = 0 \quad \text{pour } i=1, \dots, m \quad (1.6)$$

Le choix $z_0 = \max\{-q_i ; 1 \leq i \leq m\}$ fournit alors la solution du système (P_3) donnée par

$$z=0 \text{ et } w=q+1z_0.$$

Afin d'obtenir une solution de base réalisable complémentaire du système (P_2) lorsque celle-ci existe, la logique suivie consiste à rendre égale à zéro la variable artificielle z_0 par une série de pivotages, qu'on définit ultérieurement, en résolvant le système (P_3) .

DEFINITION 2 :

Une solution du système (P_3) est dite solution de base réalisable presque complémentaire si :

- 1) (w, z, z_0) est une solution de base réalisable de (1.4)-(1.5).
- 2) Ni w_s ni z_s se sont des variables de base pour $s \in \{1, \dots, m\}$.
- 3) z_0 est une variable de base, et pour tout $i \in \{1, \dots, m\}$ tel que $i \neq s$, soit w_i soit z_i est une variable de base.

Le plus ancien algorithme de résolution du problème PLC est la Méthode de Pivotage Principal proposé par R.W.COTTLE et G.B.DANTZIG[2]. Par la suite C.E.LEMKE[4] proposa un algorithme qui utilise la notion de pivots complémentaires. La logique de l'algorithme de C.E.LEMKE est plus simple que celle de R.W.COTTLE et G.B.DANTZIG .

DEFINITION 3 :

Soit $S = \{x \in \mathbb{R}^n ; Ax = b, x \geq 0\}$, où A une matrice d'ordre $m \times n$ et de rang m . Par définition, un vecteur d non nul est dit une direction de S si $x + \lambda d \in S$ pour tout $x \in S$ et pour tout réel $\lambda \geq 0$.

De même un vecteur d^* non nul est dit une direction extrême de S si et seulement si A peut être décomposée en $[B, N]$ de sorte que $B^{-1}a_j \leq 0$ pour certaines colonnes a_j de N , et d^* est un multiple positif de $d = (-B^{-1}a_j, e_j)^t$, où e_j est le vecteur d'ordre $(n-m)$ dont tous les éléments sont nuls à l'exception du $j^{\text{ème}}$ qui vaut 1.

IV.3.2. DESCRIPTION DE L'ALGORITHME DE LEMKE :

Initialisation:

si $q \geq 0$, stop; $(w, z) = (q, 0)$ est une solution de base réalisable complémentaire. Sinon, il existe $i \in \{1 \dots m\}$ tel que $q_i < 0$, et l'on présente le problème (1.4)-(1.5) sous forme d'un tableau (analogue au tableau du simplexe).

Soit $q_s = \min\{q_i ; \text{pour } i=1 \dots m\}$, on met à jour le tableau par pivotage en ligne s et colonne z_0 . w_s quitte la base et z_0 rentre dans la base. Ainsi pour $i=1 \dots m$ avec $i \neq s$ z_0 et w_i sont positifs ou nuls. Posons alors $y_s = z_s$ et passons à l'étape principale.

Etape principale:

Cette étape se décompose en quatre phases :

Phase 1 : Soit d_s la colonne qui correspond à la variable y_s dans le tableau courant.

Si $d_s \leq 0$ on va en 4). Sinon, on détermine un indice r tel que :

$$\frac{q_r}{d_{s_r}} = \min\left\{\frac{q_i}{d_{s_i}} ; d_{s_i} > 0 ; 1 \leq i \leq m\right\}$$

Le vecteur q^* définit la colonne représentant le du second membre de l'équation (1.4) (noté dans le tableau s.m) après mise à jour du tableau. Si la variable de base correspondant à la ligne r est z_0 , aller en 3). sinon aller en 2).

Phase 2 : La variable de base correspondant à la ligne r est soit w_k soit z_k (pour une composante $k \neq s$). La variable y_s rentre dans la base et le tableau se définit par pivotage en ligne r et colonne y_s . Si la variable qui quitte la base est w_k (resp z_k) on pose alors $y_s = z_k$ (resp $y_s = w_k$) et on retourne en 1).

Phase 3 : La variable y_s (c'est à dire z_s) rentre dans la base, et z_0 quitte la base, et l'on opère le pivotage en colonne y_s et ligne z_0 . On obtient ainsi une solution de base réalisable complémentaire.

Phase 4 : On s'arrête avec un rayon de terminaison. Le rayon $R = \{(w, z, z_0) + \lambda d ; \lambda \in \mathbb{R} \text{ et } \lambda \geq 0\}$ est déterminé de telle sorte que tout point de R satisfasse au système (P_3) . (w, z, z_0) est une solution de base réalisable presque complémentaire associée au dernier tableau et d est une direction extrême dont les composantes valent 1 pour la ligne correspondant à y_s , $-d_s$ pour les lignes correspondant aux variables de base et 0 ailleurs.

L'algorithme se termine ainsi.

Pour plus détails le lecteur peut se référer aux travaux de R.W.COTTLE and G.B.DANTZIG[2], C.E.LEMKE[3], [4], O.L.MANGAZARIAN[5], [6], J.A.TOMLIN[9], A.YASSINE[10] et M.S.BAZARAA and C.M.SHETTY[1].

IV.3.3. EXEMPLE (POUR LEQUEL L'ALGORITHME CONVERGE VERS UNE SOLUTION REALISABLE)

On veut déterminer une solution du problème linéaire complémentaire PLC défini par :

$$M = \begin{bmatrix} 0 & 0 & -1 & -1 \\ 0 & 0 & 1 & -2 \\ 1 & -1 & 2 & -2 \\ 1 & 2 & -2 & 4 \end{bmatrix} \quad q = \begin{bmatrix} 2 \\ 2 \\ -2 \\ -6 \end{bmatrix}$$

Etape d'initialisation : On introduit la variable auxiliaire z_0 et on construit le tableau suivant :

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	z_0	s.m
w_1	1	0	0	0	0	0	1	1	-1	2
w_2	0	1	0	0	0	0	-1	2	-1	2
w_3	0	0	1	0	-1	1	-2	2	-1	-2
w_4	0	0	0	1	-1	-2	2	-4	(-1)	-6

On observe alors que $\min\{q_i : 1 \leq i \leq 4\} = q_4$. On entoure donc l'élément qui se trouve à l'intersection de la colonne z_0 et de la quatrième ligne dans laquelle figure la variable w_4 . Cet élément s'appelle le pivot. On fait alors le pivotage en ligne 4 et en colonne z_0 . on passe à l'itération 1 avec $y_s = z_4$.

Itération 1 :

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	z_0	s.m
w_1	1	0	0	-1	1	2	-1	5	0	8
w_2	0	1	0	-1	1	2	-3	6	0	8
w_3	0	0	1	-1	0	3	-4	6	0	4
z_0	0	0	0	-1	1	2	-2	4	1	6

$y_s=z_4$ rentre dans la base. Le quotient défini dans la phase 1) de l'étape principale de l'algorithme atteint son minimum pour l'indice 3. Ainsi w_3 quitte la base et l'on pose $y_s=z_3$. On effectue alors le pivotage en ligne w_3 et colonne z_4 et l'on passe à l'itération 2.

Itération 2:

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	z_0	s.m
w_1	1	0	-5/6	-1/6	1	-1/2	7/3	0	0	14/3
w_2	0	1	-1	0	1	-1	1	0	0	4
z_4	0	0	1/6	-1/6	0	1/2	-2/3	1	0	2/3
z_0	0	0	-2/3	-1/3	1	0	2/3	0	1	10/3

$y_s=z_3$ rentre dans la base. Le quotient défini dans la phase 1) de l'étape principale de l'algorithme atteint son minimum pour l'indice 1. Ainsi w_1 quitte la base et z_3 rentre dans la base et on pose $y_s=z_1$ puis l'on passe à l'itération 3 :

Itération 3

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	z_0	s.m
z_3	3/7	0	-5/14	-1/14	3/7	-3/14	1	0	0	2
w_2	-3/7	1	-9/14	-1/14	4/7	-11/14	0	0	0	2
z_4	2/7	0	-1/14	-3/14	2/7	5/14	0	1	0	2
z_0	-2/7	0	-3/7	-2/7	5/7	1/7	0	0	1	2

$y_s=z_1$ rentre dans base. Le quotient défini dans la phase 1) de l'étape principale de l'algorithme atteint son minimum pour l'indice correspondant à la ligne de z_0 . Ainsi z_0 quitte la base et z_1 rentre dans la base. On obtient ainsi une solution de base réalisable complémentaire donnée par le

tableau suivant :

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	z_0	s.m
z_3	3/5	0	-1/10	1/10	0	-3/10	1	0	-3/5	4/5
w_2	-1/5	1	-3/10	3/10	0	-9/10	0	0	-4/5	2/5
z_4	2/5	0	1/10	-1/10	0	3/10	0	0	-2/5	6/5
z_1	-2/5	0	-3/5	-2/5	1	1/5	0	0	7/5	14/5

En résumé, l'algorithme de pivotage complémentaire produit le point

$$(w_1, w_2, w_3, w_4, z_1, z_2, z_3, z_4) = (0, 2/5, 0, 0, 14/5, 0, 4/5, 6/5)$$

IV.3.4. FINITUDE DE LA CONVERGENCE DE LA METHODE DE LEMKE :

L'algorithme de C.E.LEMKE converge en un nombre fini d'itérations vers une solution de base réalisable complémentaire lorsqu'elle existe. Sinon il s'arrête avec un rayon de terminaison. Mais sous certaines conditions portant sur la matrice M , on vérifie que l'algorithme produit toujours une solution de base réalisable après un nombre fini d'itérations.

DEFINITION 1 :

Une matrice M d'ordre $n \times n$ est dite copositive si pour tout $z \geq 0$ on a $z^T M z \geq 0$. M est copositive-plus si M est copositive et de plus si $z \geq 0$ et $z^T M z = 0$ impliquent que $(M + M^T)z = 0$.

2.9) THEOREME [] : *Sous les hypothèses de non-dégénérescence et de copositivité plus de M , l'algorithme de pivotage complémentaire converge en un nombre fini d'itérations. En particulier, si le système défini par (1.1)-(1.2) est consistant, l'algorithme produit une solution de base réalisable complémentaire. De plus, lorsque ce système est non-consistant, l'algorithme produit un rayon de terminaison.*

Preuve: voir M.S.BAZARAA[1].

IV.4. REDUCTION DU PROBLEME DE PROGRAMMATION QUADRATIQUE EN UN PROBLEME PLC:

Le problème posé en programmation quadratique s'exprime comme la recherche du minimum d'une fonction quadratique sous contraintes linéaires. Ainsi, le problème de programmation quadratique qu'on note (P_4) consiste à minimiser

$$f(x) = 1/2 \langle x, Cx \rangle + \langle x, d \rangle$$

sous les contraintes

$$\langle a_i, x \rangle \leq b_i \quad i=1, \dots, m.$$

Avec $x \in \mathbb{R}^n$, $a_i \in \mathbb{R}^n$, $d \in \mathbb{R}^n$, $b_i \in \mathbb{R}$ et C est une matrice d'ordre $n \times n$ symétrique définie positive. Posons $A = [a_1, a_2, \dots, a_m]^t$.

PROPOSITION 1 : Le problème (P_4) est équivalent au problème linéaire complémentaire (P_5) suivant :

Trouver $z \in \mathbb{R}^m$, $\lambda \in \mathbb{R}^m$ tels que

$$z - M\lambda = q;$$

$$z \geq 0, \lambda \geq 0 \text{ et}$$

$$\langle z, \lambda \rangle = 0$$

avec $M = AC^{-1}A'$, $q = AC^{-1}d$ et $z = -Ax + b$.

Preuve :

On note par λ le vecteur multiplicateur de Lagrange associé aux contraintes $Ax \leq b$ et on pose $z = -Ax + b$. Les conditions de KUHN-TUCKER associés au problème (P_4) s'écrivent :

$$Ax - b \leq 0, \quad Cx + d + A'\lambda = 0 \quad \text{et} \quad \langle Ax - b, \lambda \rangle = 0.$$

On obtient par prémultiplication de la première équation par C^{-1}

$$x + C^{-1}A'\lambda = -C^{-1}d \quad (1)$$

La relation (1) donne :

$$x = -C^{-1}A'\lambda - C^{-1}d \quad (2)$$

D'où

$$z = -Ax + b = -A(-C^{-1}A'\lambda - C^{-1}d) + b = AC^{-1}A'\lambda + AC^{-1}d + b. \quad (3)$$

En posons $M = AC^{-1}A'$, $q = AC^{-1}d + b$, alors la relation (3) s'écrit :

$$z - M\lambda = q. \quad (4)$$

Les conditions de KUHN-TUCKER du problème (P_4) se traduisent ainsi :

Trouver $z \in \mathbb{R}^m$, $\lambda \in \mathbb{R}^m$ tels que

$$z - M\lambda = q;$$

$$z \geq 0, \lambda \geq 0 \text{ et}$$

$$\langle z, \lambda \rangle = 0$$

avec $M = AC^{-1}A'$, $q = AC^{-1}d + b$ et $z = -Ax + b$.

Lemme 1 :

Si les vecteurs $a_i, i=1, \dots, m$ sont linéairement indépendants la matrice AA' est régulière.

Preuve : voir Lemme 1 du paragraphe II.4.3. du chapitre II.

PROPOSITION 2 : Si (z^*, λ^*) est solution du problème PLC (P_5) alors $(-C^{-1}A'\lambda^* - C^{-1}d)$ est solution du problème (P_4).

Si x^* est solution du problème (P_4) alors $(-Ax^* + b, -(AA')ACx^* - (AA')^{-1}Ad)$ est solution du problème (P_5).

Preuve : voir proposition du paragraphe II. chapitre II.

PROPOSITION 3 :

La matrice $M = AC^{-1}A'$ qui définit le problème linéaire complémentaire (Q) est une matrice copositive-plus.

Preuve :

Pour cela il faut montrer que $M=AC^{-1}A^t$ est copositive et si $z \geq 0$ et $z^t M z = 0$, alors $(M+M^t)z=0$.

1) $M=AC^{-1}A^t$ copositive :

en effet, on a pour tout $z \geq 0$

$$\langle z, AC^{-1}A^t z \rangle = \langle A^t z, C^{-1}A^t z \rangle \geq 0$$

car C^{-1} est définie positive ce qui traduit la copositivité de la matrice $M=AC^{-1}A^t$.

2) montrons maintenant que :

$$z \geq 0, z^t (AC^{-1}A^t) z = 0 \Rightarrow (M+M^t)z = [AC^{-1}A^t + AC^{-1}A^t]z = 0.$$

La matrice $A^t C^{-1} A$ est symétrique et donc

$$[AC^{-1}A^t + (AC^{-1}A^t)^t]z = 2(AC^{-1}A^t)z = 2AC^{-1}(A^t z)$$

De plus, puisque C^{-1} est définie positive

$$z^t (A^t C^{-1} A) z = 0 \Leftrightarrow A^t z = 0 \quad (2)$$

Ce qui montre que la matrice $AC^{-1}A^t$ est copositive-plus.

COROLLAIRE 1 : *Supposons la région d'admissibilité du problème (P_4) non vide. Sous l'hypothèse de non dégénérescence, l'algorithme de LEMKE produit une solution du problème linéaire complémentaire (P_5) associé ou un point de KUHN-TUCKER en un nombre fini d'itérations.*

Preuve :

D'après le THEOREME 1 du paragraphe III.3.4. et la PROPOSITION 3 du paragraphe III.4.

Remarque :

Le problème **PLC** est un cas particulier du problème monotone complémentaire étudié au chapitre 1) caractérisé par un opérateur monotone maximal T donné par

$$T(\lambda) = AC^{-1}A^t \lambda + AC^{-1}d.$$

IV.5. APPLICATIONS:

IV.5.1. LE PROBLEME DE REGRESSION ISOTONE AU SENS DES MOINDRES CARRES :

L'étude faite au chapitre II permet d'écrire le problème linéaire complémentaire associé à un problème de régression isotone dans la forme :

Trouver $z \in \mathbb{R}^{n-1}$ et $\lambda \in \mathbb{R}^{n-1}$ solutions de :

$$z - AW^{-1}A^t\lambda = -AW^{-1}g$$

$$(P_6) \quad z \geq 0, \lambda \geq 0 \text{ et}$$

$$\langle z, \lambda \rangle = 0.$$

La matrice $AW^{-1}A^t$ est donnée en page 64 chapitre II. Ainsi le problème de la régression isotone peut être résolu par la méthode de C.E.LEMKE décrite ci-dessus.

La matrice $(AW^{-1}A^t)$ est copositive-plus, puisque W est symétrique définie positive (voir PROPOSITION 3 du paragraphe III.4.), par suite la méthode converge vers une solution de base réalisable complémentaire (z^*, λ^*) après un nombre fini d'itérations.

Une fois λ^* déterminé on déduit de la PROPOSITION 2 du paragraphe III.4 du chapitre II. la solution du problème de régression isotone :

$$x^* = -W^{-1}A^t\lambda^* + W^{-1}g.$$

Tout calcul fait la valeur explicite de x^* est égale à:

$$x^* = \left(\frac{g_1 - \lambda^*_1}{w_1}, \frac{\lambda^*_1 g_2 - \lambda^*_2}{w_2}, \dots, \frac{\lambda^*_{n-2} g_{n-1} - \lambda^*_{n-1} g_n + \lambda^*_n}{w_{n-1}}, \frac{g_n + \lambda^*_n}{w_n} \right)$$

On suit le même raisonnement Pour la résolution du problème de *la régression concave* par la méthode de C.E.LEMKE. Seule la matrice qui définit le système (P_6) change. Elle est donnée en page 65 chapitre II.

REFERENCE

- [1] **M.S.BAZARAA and C.M.SHETTY(1972)**
Non Linear Programming.
- [2] **R.W.COTTLE and G.B.DANTZIG(1968)**
Complementary Pivot Theory of Mathematical Programming.
Am. Math. Soc. Providence, pp. 115-138.
- [3] **C.E.LEMKE(1965)**
Bimatrix Equilibrium Points and Mathematical Programming.
Management Science 4, pp. 681-689.
- [4] **C.E.LEMKE(1969)**
Recent Results on Complementarity Problems.
in J.B.Rosen, O.L.Mangazarian and K.Ritter, eds., Academic Press New York, Non Linear Programming, pp. 349-384.
- [5] **O.L.MANGAZARIAN(1969)**
Non Linear Programming.
Mc Graw-Hill, New York.
- [6] **O.L.MANGAZARIAN(1985)**
Simple Computable Bounds for Solutions of Linear Complementarity Problems and Linear Programs.
Mathematical Programming 25, pp. 1-12.
- [7] **M.MINOUX(1983)**
Programmation mathématique : théorie et algorithmes.
Dunod, Paris.
- [8] **M.SAKAROVITCH(1982)**
Linear Programming.
Springer-Verlag, New York.
- [9] **J.A.TOMLIN(1978)**
Robust Implementation of LEMKE's Method for the Linear Complementary Problem. Mathematical Programming Study, 7, pp. 55-60.
- [10] **A.YASSINE(1986)**
Méthode de pivotage de LEMKE et ses applications.
Rapport de D.E.A., IMAG, Grenoble, juin 1985.

CHAPITRE IV

***RESOLUTION DES PROBLEMES DE REGRESSION PAR LA
METHODE DU GRADIENT CONJUGUE.***



IV.1. INTRODUCTION

Le problème de la *régression isotone* et celui de la *régression concave* se ramènent, après une simple transformation, à des problèmes de *programmation quadratique*. Cette classe de problèmes se propose pour objet l'étude théorique du problème de la minimisation d'une fonction quadratique avec ou sans contraintes, ainsi que la conception et la mise en œuvre des algorithmes de résolution. Elle est très intéressante par son champ d'applications que ce soit dans les sciences de l'ingénieur, ou dans d'autres domaines des mathématiques appliquées. Elle a suscité jusqu'à présent un volume considérable de publications ([3], [5], [7], [9], [13], [15]).

Dans ce chapitre et en un premier temps, on présente le problème de la programmation quadratique dans un cadre théorique général. *La méthode des directions conjuguées*, destinée à résoudre ce type de problème, est passée en revue et on insiste plus particulièrement sur une de ses variantes, connue en littérature sous le nom de *méthode du gradient conjugué* ([7], [14], [15]). Il faut signaler que cette classe de méthodes a la particularité de converger en *un nombre fini d'itérations*.

Dans le souci de simplifier et de faciliter la résolution des problèmes de régression et via la *théorie de dualité*, on forme le problème dual d'un programme quadratique. Le passage au problème dual nous ramène à un problème du même type que le problème primal, i.e. un problème de programmation quadratique. D'autre part, le problème dual présente certains avantages sur le primal tels que la *diminution de la dimension de l'espace des variables et la simplicité des contraintes (contraintes de positivité)*, ce qui parfois, simplifie considérablement l'algorithme de résolution. Ainsi, *la méthode du gradient conjugué* pour la résolution du problème dual serait d'un emploi particulièrement *simple et commode*.

A la fin de ce chapitre, on expose les deux problèmes de régression, en les formulant en termes de programmation quadratique moyennant une simple opération. Le dual des deux problèmes sera aussi présenté.

IV.2. PROBLEME DE PROGRAMMATION QUADRATIQUE :

On entend par problème de programmation quadratique, la recherche du minimum d'une fonction quadratique avec des contraintes linéaires. Ainsi, le problème de programmation quadratique consiste à minimiser

$$f(x) = 1/2 \langle x, Cx \rangle + \langle x, d \rangle, \tag{1.1}$$

sous contraintes

$$Ax \leq b, \tag{1.2}$$

où

n = nombre de variables.

m = nombre de contraintes.

C = matrice réelle $n \times n$ symétrique définie positive.

A = matrice réelle $m \times n$ (matrice des contraintes) définie par $A = [a_1, a_2, \dots, a_m]^t$

avec $a_i \in \mathbb{R}^n$ pour $1 \leq i \leq m$, et $x \in \mathbb{R}^n$, $d \in \mathbb{R}^n$, $b \in \mathbb{R}^m$.

Ce type de problème est très intéressant par son domaine d'applications. Il a fait l'objet de plusieurs travaux; on cite entre autres P. WOLFE[3], A. LENT and Y. CENSOR[13], C. HILDRETH[9] et G. ZOUTENDIJK[16].

IV.2.1. CONDITIONS NECESSAIRES ET SUFFISANTES D'OPTIMALITE :

On notera D l'ensemble des solutions réalisables du problème (1.1)-(1.2) c'est à dire :

$$D = \{x \in \mathbb{R}^n ; \langle a_i, x \rangle \leq b_i, i=1, \dots, m\},$$

posons $I = \{1, 2, \dots, m\}$.

Le théorème suivant est fondamental et donne une condition nécessaire et suffisante d'optimalité globale pour le problème (1.1) sous les contraintes du type (1.2).

THEOREME 1 (KUHN-TUCKER) [9] : Une condition nécessaire et suffisante pour que $x^* \in D$ soit un optimum de (1.1)-(1.2) est qu'il existe des nombres $\lambda_i \geq 0$, pour $i=1, \dots, m$, tels que

$$Cx^* + d + \sum_{i=1}^m \lambda_i a_i = 0$$

$$\lambda_i [\langle a_i, x^* \rangle - b_i] = 0 \quad \forall i \in I.$$

DEFINITION 1 : Les nombres λ_i figurant dans le théorème s'appellent les multiplicateurs de Lagrange.

IV.2.2. METHODES DES DIRECTIONS CONJUGUEES :

PRINCIPE GENERAL :

Il s'agit des méthodes itératives qui, appliquées à une fonction quadratique de n variables, conduisent à l'optimum en au plus n étapes.

Considérons une fonction quadratique :

$$f(x)=1/2\langle x,Cx\rangle+\langle x,d\rangle$$

Le principe *des méthodes de directions conjuguées* consiste à partir d'un point, à minimiser f(x) successivement suivant n directions linéairement indépendantes d_0, d_1, \dots, d_{n-1} , possédant la propriété d'être mutuellement conjuguées par rapport à la forme quadratique f(x). Rappelons que cette propriété s'exprime par :

$$\left. \begin{array}{l} \forall i \text{ tel que } 1 \leq i \leq n-1 \\ \forall j \text{ tel que } 1 \leq j \leq n-1 \\ i \neq j \end{array} \right\} \Rightarrow d_i^t \cdot C \cdot d_j = 0$$

Une méthode de directions conjuguées converge donc de façon finie en au plus n étapes dans le cas de la recherche d'un minimum d'une fonction quadratique sans contraintes.

Comme il existe un certain nombre de degrés de liberté pour le choix des directions (d_0, d_1, \dots, d_{n-1}), on peut imaginer de nombreux algorithmes basés sur le principe général des directions conjuguées. Nous présentons plus particulièrement la méthode dite du *gradient conjugué conditionnel*.

IV.2.3. METHODE DU GRADIENT CONJUGUE :

La méthode du gradient conjugué a été proposée à l'origine par HESTENES[8] pour résoudre un système d'équations linéaires, et la première application en optimisation non linéaire est due à FLETCHER & REEVES[5].

Le problème qu'on considère ici consiste à minimiser

$$f(x)=1/2\langle x,Cx\rangle+\langle x,d\rangle, \tag{3.1}$$

sur tout l'espace \mathbb{R}^n .

L'idée de la méthode est de construire progressivement des directions d_0, d_1, \dots, d_k mutuellement conjuguées par rapport à la matrice C de la forme quadratique.

A chaque étape k , la direction d_k est obtenue par combinaison linéaire du gradient $-\nabla f(x^k)$ en x^k , et les directions précédentes d_0, d_1, \dots, d_{k-1} , les coefficients de la combinaison linéaire étant choisis de telle sorte que d_k soit conjuguée par rapport à toutes les directions précédentes.

Afin de clarifier la présentation, on explicite cette idée comme suit :

Supposons qu'on est à l'itération k et on a un point x^k . Etant donnés $(k+1)$ vecteurs linéairement indépendants d_0, d_1, \dots, d_k engendrant un sous espace \mathcal{D}_k et posons $D_k = [d_0, d_1, \dots, d_k]$ une matrice dont les colonnes sont les $(k+1)$ vecteurs d_0, d_1, \dots, d_k .

L'itéré suivant x^{k+1} est le point qui réalise le minimum de $f(x)$ sur le translaté $x^k + \mathcal{D}_k$.

Or, minimiser $f(x)$ sur le translaté $x^k + \mathcal{D}_k$ est équivalent au problème suivant :

$$\min\{f(x^k + D_k w) \ ; \ w \in \mathbb{R}^{k+1}\}.$$

Si $x^k + D_k w$ est substitué dans (3.1), on obtient le problème suivant :

$$\min\{1/2 \langle w, D_k^t C D_k w \rangle + \langle w, D_k^t g_k \rangle \ ; \ w \in \mathbb{R}^{k+1}\}, \quad (3.2)$$

avec $g_k = \nabla f(x^k) = Cx^k + d$. Le problème (3.2) admet pour solution le point :

$$w = -(D_k^t C D_k)^{-1} D_k^t g_k$$

Par conséquent x^{k+1} est donné par :

$$x^{k+1} = x^k - D_k (D_k^t C D_k)^{-1} D_k^t g_k. \quad (3.3)$$

Cette façon de faire nous permet de voir d'abord que g_{k+1} , le gradient de $f(\cdot)$ au point x^{k+1} , est orthogonal aux vecteurs $(d_i)_{0 \leq i \leq k}$ (les colonnes de D_k). En effet,

$$\begin{aligned} D_k^t g_{k+1} &= D_k^t (Cx^{k+1} + d) = D_k^t (Cx^k + d - CD_k (D_k^t C D_k)^{-1} D_k^t g_k) \\ &= D_k^t g_k - D_k^t C D_k (D_k^t C D_k)^{-1} D_k^t g_k = D_k^t g_k - D_k^t g_k = 0 \end{aligned}$$

Ensuite, si chaque itéré x^j , $j=1, \dots, k$, est obtenu en minimisant $f(\cdot)$ sur le translaté $x^{j-1} + \mathcal{D}_{j-1}$, alors par récurrence il vient, pour $j=1, \dots, k$

$$\langle g_j, d_i \rangle = 0 \text{ pour } j > i, \quad (3.4)$$

l'expression (3.3) devient alors

$$x^{k+1} = x^k + \delta_k D_k (D_k^T C D_k)^{-1} e_k, \quad (3.5)$$

où e_k est la $k^{\text{ième}}$ colonne de la matrice identité et $\delta_k = -\langle g_k, d_k \rangle$.

Supposons que les $(k+1)$ vecteurs sont mutuellement conjugués par rapport à la matrice C , i.e.

$$\left. \begin{array}{l} \forall i \text{ tel que } 0 \leq i \leq k \\ \forall j \text{ tel que } 0 \leq j \leq k \\ i \neq j \end{array} \right\} \Rightarrow d_i^T \cdot C \cdot d_j = 0 \quad (3.6).$$

Si (3.6) est vérifiée, (3.5) devient

$$x^{k+1} = x^k + \lambda^k d_k \quad (3.7).$$

$$\text{Avec } \lambda^k = - \frac{\langle g_k, d_k \rangle}{\langle d_k, C d_k \rangle} = \frac{\|g_k\|^2}{\langle d_k, C d_k \rangle} > 0.$$

La valeur de λ^k est le pas de déplacement. De plus, (3.7) indique que le $(k+1)^{\text{ième}}$ vecteur d_k est une direction de déplacement pour la fonction $f(\cdot)$.

Par définition de $f(\cdot)$ on a

$$g_{i+1} - g_i = C(x^{i+1} - x^i) = \lambda^i C d_i. \quad (3.8)$$

Posons $y_i = g_{i+1} - g_i$ on aura alors

$$\langle y_i, d_j \rangle = \lambda^i \langle C d_i, d_j \rangle = \lambda^i d_i^T \cdot C \cdot d_j = 0 \text{ pour } i \neq j. \quad (3.8')$$

Maintenant, on s'intéresse au calcul des directions d_i , $i=0,1,\dots,k$. Posons $d_0 = -g_0 = -C x^0 - d$; à chaque étape k la direction d_k est obtenue par combinaison linéaire de g_k , et les directions précédentes d_0, d_1, \dots, d_{k-1} ,

$$d_k = -g_k + \sum_{j=1}^{k-1} \beta_{k,j} d_j \quad (3.9).$$

de (3.9) on voit facilement que g_k est combinaison linéaire des vecteurs d_0, d_1, \dots, d_k , par conséquent $g_i \in \mathcal{D}_k$ pour $1 \leq i \leq k$. Comme $d_0 = -g_0$, alors $g_0 \in \mathcal{D}_k$

Par conséquent, d'après (3.4) on a

$$\langle g_k, g_i \rangle = 0 \quad k > i. \quad (3.10)$$

Prémultiplions (3.9) par $d_i^t C$ et en tenant compte de (3.6) et (3.8) on obtient pour $i=0, \dots, k-1$:

$$d_i^t C \cdot d_k = d_i^t C \cdot g_k + \sum_{j=0}^{k-1} \beta_{k,j} d_i^t C \cdot d_j = -\frac{1}{\lambda} (g_{i+1} - g_i)^t g_k + \beta_{k,i} d_i^t C \cdot d_i \quad (3.11).$$

La relation (3.10) implique que le premier terme du second membre de la relation (3.11) s'annule pour $k-1 > i$; ainsi et pour avoir d_k conjugué à d_i pour $k-1 > i$ il suffit de choisir $\beta_{k,i} = 0$ pour $k-1 > i$.

Alors $\beta_{k,k-1}$ est le seul coefficient différent de zéro, et pour ne pas alourdir le texte, en notera dorénavant ce coefficient par β_k . Pour déterminer la valeur de β_k tel que d_k soit conjugué à d_{k-1} , on prémultiplie (3.9) par y_{k-1}^t et en tenant compte du fait que $\langle y_{k-1}, d_k \rangle = 0$, on obtient alors

$$0 = -\langle y_{k-1}, g_k \rangle + \beta_{k-1} \langle y_{k-1}, d_{k-1} \rangle,$$

d'où

$$\beta_{k-1} = \frac{\langle y_{k-1}, g_k \rangle}{\langle y_{k-1}, d_{k-1} \rangle} \quad (3.12).$$

par conséquent, d_k peut être écrit ainsi

$$d_k = -g_k + \beta_{k-1} d_{k-1}, \quad (3.13)$$

où β_{k-1} est donnée par (3.12).

Compte tenu de (3.10) on a

$$\langle y_{k-1}, g_k \rangle = \langle g_k - g_{k-1}, g_k \rangle = \langle g_k, g_k \rangle - \langle g_k, g_{k-1} \rangle = \langle g_k, g_k \rangle,$$

et

$$\langle y_{k-1}, d_{k-1} \rangle = \langle g_k - g_{k-1}, -g_{k-1} + \beta_{k-2} d_{k-2} \rangle = \langle g_k, g_{k-1} \rangle + \langle g_{k-1}, g_{k-1} \rangle + \beta_{k-2} y_{k-1}^t d_{k-2}$$

D'après (3.8') $y_{k-1}^t d_{k-2} = 0$ et d'après (3.10) $\langle g_k, g_{k-1} \rangle = 0$, d'où

$$\langle y_{k-1}, d_{k-1} \rangle = \langle g_{k-1}, g_{k-1} \rangle = \|g_{k-1}\|^2$$

Par conséquent, β_{k-1} prend la valeur suivante :

$$\beta_{k-1} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} = \frac{\|\nabla f(x^k)\|^2}{\|\nabla f(x^{k-1})\|^2}$$

En posant $g_k = \nabla f(x^k)$, la méthode du gradient conjugué se schématise par :

Algorithme du gradient conjugué pour les fonctions quadratiques :

- a) Soient x^0 le point de départ et $d_0 = -g_0 = -\nabla f(x^0) = -Cx^0 - d$
- b) A l'itération k , on est au point x^k , et on détermine x^{k+1} par $x^{k+1} = x^k + \lambda^k d_k$ avec

$$(1) \lambda^k = - \frac{\langle g_k, d_k \rangle}{\langle d_k, Cd_k \rangle} = \frac{\|g_k\|^2}{\langle d_k, Cd_k \rangle} \quad \text{puis}$$

$$(2) \quad d_{k+1} = -g_{k+1} + \beta_k d_k \quad \text{avec}$$

$$(3) \quad \beta_k = \frac{\langle g_{k+1}, Cd_k \rangle}{\langle d_k, Cd_k \rangle}$$

Faire $k \leftarrow k+1$ et retourner en (b).

THEOREME 1 (N.Gastinel [6]):

La méthode du gradient conjugué décrite ci-dessus converge en n itérations au plus.

Le principe est formé sur la remarque : les g_i successifs vont en s'orthogonalisant, c'est à dire qu'on est obligé de tomber sur un g_k égal à 0.

IV.2.4. IDEE GENERALE SUR LA RESOLUTION NUMERIQUE DU PROBLEME (1.1)-(1.2) :

La résolution numérique du problème (1.1)-(1.2) se base sur la méthode du gradient conjugué décrite ci-dessus et dont l'idée principale est la suivante :

Soit x^0 un point admissible, i.e. $Ax^0 \leq b$. Les contraintes saturées définissent une face d'un ensemble polyédrique décrit par les inégalités linéaires $Ax \leq b$.

On résoud le problème qui consiste à minimiser $f(x)$ sur cette face par la méthode du gradient conjugué.

Deux éventualités peuvent se présenter. La première est que le point obtenu est solution du problème proposé (1.1)-(1.2). La deuxième éventualité est de passer à une autre face et la procédure reprend par le commencement. Vu que la méthode du gradient conjugué minimise $f(x)$ sur chaque face en un nombre fini d'itérations et que les faces sont en nombre fini, un tel algorithme converge au bout d'un nombre fini de pas.

IV.2.5. OPERATEUR DE PROJECTION :

Soit $I = \{i : 1 \leq i \leq m\}$ et \mathcal{J} un sous ensemble de I .

Formons une matrice $A_{\mathcal{J}}$ dont les lignes sont les vecteurs $a_i, i \in \mathcal{J}$, de sorte que c'est une matrice $p \times n$, p étant le nombre d'éléments de \mathcal{J} .

Lemme 1 : Les vecteurs $a_i, i \in \mathcal{J}$, sont linéairement indépendants si et seulement si la matrice $A_{\mathcal{J}}A_{\mathcal{J}}^t$ est régulière.

Preuve :

voir **Lemme 1** du paragraphe II.4.3 du chapitre II.

Posons

$$P = A_{\mathcal{J}}^t (A_{\mathcal{J}} A_{\mathcal{J}}^t)^{-1} A_{\mathcal{J}}$$

C'est facile de vérifier les propriétés suivantes de l'opérateur P :

- 1) $P \cdot P = P$
- 2) $P^t = P$
- 3) $P(I - P) = (I - P)P = 0$.

L'opérateur P est un opérateur de projection orthogonale dans un sous espace engendré par les vecteurs $a_i, i \in \mathcal{J}$.

IV.2.6. MINIMISATION D'UNE FONCTION QUADRATIQUE SUR UNE VARIETE AFFINE :

Admettons qu'on ait à minimiser la fonction quadratique donnée par :

$$f(x) = 1/2 \langle x, Cx \rangle + \langle x, d \rangle, \quad (6.1)$$

sous les contraintes

$$\langle a_i, x \rangle - b_i = 0 \quad i \in \mathcal{J}, \quad (6.2)$$

où \mathcal{J} est un sous-ensemble de $\{1, 2, \dots, m\}$. Les vecteurs $a_i, i \in \mathcal{J}$, sont supposés linéairement indépendants.

De ce fait, la matrice $A_{\mathcal{J}} A_{\mathcal{J}}^t$ serait régulière et on définit alors la projection orthogonale sur le sous espace engendré par les vecteurs $a_i, i \in \mathcal{J}$, par :

$$P = A_{\mathcal{J}} (A_{\mathcal{J}} A_{\mathcal{J}}^t)^{-1} A_{\mathcal{J}}^t.$$

Soit x^0 un point vérifiant (6.2); Introduisons une nouvelle variable y par la formule :

$$x = x^0 + (I - P)y, \quad (6.3)$$

et considérons la fonction quadratique

$$\varphi(y) = f(x^0 + (I - P)y).$$

Il est facile de voir que :

$$\nabla \varphi(y) = (I - P) \nabla f(x). \quad (6.4)$$

Il est facile de voir que le problème (6.1)-(6.2) se ramène à la minimisation de la fonction quadratique $\varphi(y)$ en l'absence de contrainte. La recherche du minimum de $\varphi(y)$ se fera par la méthode du gradient conjugué décrite au paragraphe IV.2.3.

Lemme 1 [15] : Soit y^* un point de minimum libre de la fonction $\varphi(y)$. Le point correspondant

$$x^* = x^0 + (I - P)y^*$$

réalise alors le minimum de la fonction $f(x)$ sous les contraintes (6.2).

La méthode du gradient conjugué pour minimiser $\varphi(y)$ est donnée par le processus itératif suivant (voir paragraphe IV.2.3) :

$$y^0=0, \quad d_0 = -\nabla\varphi(0).$$

$$y^{k+1} = y^k + \lambda^k d_k \quad \text{avec}$$

$$\lambda^k = -\frac{\langle \nabla\varphi(x^k), d_k \rangle}{\langle d_k, (I-P).C.(I-P).d_k \rangle}$$

$$d_{k+1} = -\nabla\varphi(y^{k+1}) + \beta_k d_k \quad \text{avec}$$

$$\beta_k = \frac{\|\nabla\varphi(x^{k+1})\|^2}{\|\nabla\varphi(x^k)\|^2}$$

Car la matrice qui définit le terme quadratique de $\varphi(y)$ est $(I-P).C.(I-P)$. Les formules ci-dessus définissent le processus par rapport aux variables auxiliaires. Il y a cependant intérêt à passer aux variables de départ.

La formule (6.3) entraîne

$$x^{k+1} = x^0 + (I-P)y^{k+1} \quad \forall k \geq 0, \tag{6.5}$$

or

$$y^{k+1} = y^k + \lambda^k d_k, \quad \text{d'où}$$

$$x^{k+1} = x^0 + (I-P)(y^k + \lambda^k d_k) = x^0 + (I-P)y^k + \lambda^k (I-P)d_k. \tag{6.6}$$

D'après la relation (6.5), on a

$$x^k = x^0 + (I-P)(y^k),$$

et la relation (6.6) devient :

$$x^{k+1} = x^k + \lambda^k (I-P)d_k. \tag{6.7}$$

Montrons alors par récurrence que

$$(I-P)d_k = d_k \quad \forall k \geq 0. \tag{6.8}$$

En effet, pour $k=0$ on a $(I-P)d_0 = -(I-P)\nabla\varphi(0) = -(I-P)(I-P)\nabla f(x^0)$

Or $(I-P)(I-P)=I-P-P(I-P)=I-P-P+P^2=I-P$

D'où $(I-P)d_0=-(I-P)\nabla f(x^0)=-\nabla\varphi(0)=d_0$

Supposons la relation (6.8) vraie pour k et démontrons-la pour $k+1$:

$$\begin{aligned} (I-P)d_{k+1} &= -(I-P)\nabla\varphi(y^{k+1}) + \beta_k(I-P)d_k = -(I-P)(I-P)\nabla f(x^{k+1}) + \beta_k d_k \\ &= -(I-P)\nabla f(x^k) + \beta_k d_k = -\nabla\varphi(y^{k+1}) + \beta_k(I-P)d_k = d_{k+1} \end{aligned}$$

Alors la formule (6.7) devient

$$x^{k+1} = x^k + \lambda^k d_k$$

La formule du pas λ^k , en tenant compte de (6.8), se réécrit comme suit :

$$\lambda^k = -\frac{\langle \nabla f(x^k), d_k \rangle}{\langle d_k, C.d_k \rangle}$$

Compte tenu de (6.4) la formule donnant d_{k+1} se transforme ainsi :

$$d_{k+1} = -(I-P)\nabla f(x^{k+1}) + \beta_k d_k$$

avec

$$\beta^k = \frac{\|(I-P)\nabla f(x^{k+1})\|^2}{\|(I-P)\nabla f(x^k)\|^2}$$

On peut citer alors le théorème :

THEOREME 1 : *Connaissant un point x^0 vérifiant (6.2) le problème de minimisation de la fonction (6.1) sous les contraintes (6.2) est résolu en un nombre fini de pas par le processus*

$$d_0 = -(I-P)\nabla f(x^0)$$

$$x^{k+1} = x^k + \lambda^k d_k$$

$$\lambda^k = -\frac{\langle \nabla f(x^k), d_k \rangle}{\langle d_k, C.d_k \rangle}$$

$$d_{k+1} = -(I-P)\nabla f(x^{k+1}) + \beta_k d_k, \text{ avec}$$

$$\beta^k = \frac{\|(I-P)\nabla f(x^{k+1})\|^2}{\|(I-P)\nabla f(x^k)\|^2}$$

La démonstration a été en effet effectuée plus haut lorsque nous avons déduit les formules du processus.

IV.2.7. DESCRIPTION DE L'ALGORITHME DE RESOLUTION DU PROBLEME (1.1)-(1.2) :

Reprenons le problème (1.1)-(1.2) qui consiste à minimiser

$$f(x) = 1/2 \langle x, Cx \rangle + \langle x, d \rangle, \quad (7.1)$$

sous les contraintes

$$Ax \leq b. \quad (7.2)$$

Posons $\mathcal{J}(x) = \{i; 1 \leq i \leq m \text{ tel que } \langle a_i, x \rangle - b_i = 0\}$, et supposons que pour tout x , les vecteurs $a_i, i \in \mathcal{J}(x)$, soient linéairement indépendants.

Soit x^0 un point quelconque vérifiant (1.2) et constituant l'approximation initiale. Prenons l'ensemble d'indices $\mathcal{J}_0 = \mathcal{J}(x^0)$ et construisons

$$P_{\mathcal{J}_0} = A_{\mathcal{J}_0} (A_{\mathcal{J}_0} A_{\mathcal{J}_0}^T)^{-1} A_{\mathcal{J}_0}^T$$

Calculons les quantités :

$$v_0 = -(A_{\mathcal{J}_0} A_{\mathcal{J}_0}^T)^{-1} A_{\mathcal{J}_0}^T \nabla f(x^0), \text{ et } (I - P_{\mathcal{J}_0}) \nabla f(x^0) = \nabla f(x^0) + A_{\mathcal{J}_0} v_0.$$

Deux cas peuvent se présenter:

$$1) (I - P_{\mathcal{J}_0}) \nabla f(x^0) = 0 \text{ alors } \nabla f(x^0) + A_{\mathcal{J}_0} v_0 = 0. \quad (7.3)$$

Et x_0 réalise le minimum de $f(x)$ sur une face définie par le système d'équations : $\langle a_i, x \rangle - b_i = 0 \quad i \in \mathcal{J}_0$.

Si le vecteur v_0 n'a pas de composantes $v_0^i, i \in \mathcal{J}_0$, négatives, le point x^0 est solution du problème original, car (7.3) sont alors les conditions nécessaires et suffisantes d'existence d'un minimum de $f(x)$ avec les contraintes $Ax \leq 0$.

Admettons qu'il existe un indice $j \in \mathcal{J}_0$ tel que $v_0^j < 0$.

Construisons un nouvel ensemble d'indices \mathcal{J}'_0 en éliminant de \mathcal{J}_0 l'indice j , et on résoud le problème

$$\min f(x), \quad (7.3)$$

avec

$$\langle a_i, x \rangle - b_i = 0 \quad i \in \mathcal{J}'_0 \quad (7.4)$$

par la méthode du *gradient conjugué*. Le point trouvé ne doit pas quitter le domaine admissible défini par $Ax \leq b$. Pour éviter ce phénomène, on effectue donc à chaque pas une vérification en calculant :

$$\lambda_k^* = \min \left\{ \frac{b_i - \langle a_i, x^k \rangle}{\langle a_i, d_k \rangle} ; i \in J \right\} \quad \text{où } J = \{ i ; \langle a_i, d_k \rangle > 0 \}.$$

Ici x^k est un point de la suite générée par l'algorithme et d_k la direction conjuguée en ce point. Soit maintenant λ^k la grandeur correspondant au pas de la méthode du *gradient conjugué*.

Si $\lambda^k < \lambda_k^*$ alors $x^{k+1} = x^k + \lambda^k d_k$, et le processus continue.

Si $\lambda^k \geq \lambda_k^*$ alors $x^{k+1} = x^k + \lambda_k^* d_k$, et l'algorithme s'arrête.

Conclusion : ou bien on réalise le minimum de $f(x)$ sous les conditions (7.4), ou bien le processus s'arrête quand $\lambda^k \geq \lambda_k^*$.

Dans les deux cas on initialise avec le point obtenu comme on l'a fait avec le point initial.

II) $(I - P_{J_0}) \nabla f(x^0) \neq 0$.

Dans ce cas, on résoud par la technique du *gradient conjugué*, le problème de minimisation de $f(x)$ sous les contraintes

$$(a_i, x) - b_i = 0 \quad i \in J_0, \tag{7.5}$$

avec comme point de départ x_0 . Comme dans le premier cas, on teste à chaque pas l'admissibilité des points obtenus, i.e. on calcule

$$\lambda_k^* = \min \left\{ \frac{b_i - \langle a_i, x^k \rangle}{\langle a_i, d_k \rangle} ; i \in J \right\} \quad \text{où } J = \{ i ; \langle a_i, d_k \rangle > 0 \}.$$

De même que dans le cas 1) on aura :

Si $\lambda^k < \lambda_k^*$ alors $x^{k+1} = x^k + \lambda^k d_k$, et le processus continue.

Si $\lambda^k \geq \lambda_k^*$ alors $x^{k+1} = x^k + \lambda_k^* d_k$, et l'algorithme s'arrête.

Conclusion : ou bien on réalise le minimum de $f(x)$ sous les conditions (7.5) ou bien le processus s'arrête quand $\lambda^k \geq \lambda_k^*$.

(1.1)-(1.2), l'algorithme ainsi décrit, converge en un nombre fini d'itérations

Preuve : On va faire la démonstration de ce **THEOREME** en trois phases, sous forme de **Lemmes**.

Lemme 1 : Le vecteur $d_0 = -(I - P_{J_0})\nabla f(x^0)$ est solution du problème :

$$\min_{\mathbb{R}^n} \varphi(d) = \langle \nabla f(x^0), d \rangle + 1/2 \|d\|^2,$$

avec les contraintes

$$A_{J_0} d = 0. \quad (7.6)$$

Preuve :

en effet,

$$A_{J_0} d_0 = -A_{J_0}(I - P_{J_0})\nabla f(x^0) = -A_{J_0}\nabla f(x^0) + A_{J_0}A_{J_0}^{-1}(A_{J_0}A_{J_0}^{-1})^{-1}A_{J_0}\nabla f(x^0) = 0$$

donc d_0 vérifie (7.6). De plus

$$\nabla \varphi(d_0) = d_0 + \nabla f(x^0) = -(I - P_{J_0})\nabla f(x^0) + \nabla f(x^0) = +P_{J_0}\nabla f(x^0) = -A_{J_0}^{-1}v_0,$$

où v_0 est donné par

$$v_0 = -(A_{J_0}A_{J_0}^{-1})^{-1}A_{J_0}\nabla f(x^0), \text{ d'où}$$

$$\nabla \varphi(d_0) + A_{J_0}^{-1}v_0 = 0, \quad (7.7)$$

la relation (7.7) traduit la condition nécessaire et suffisante pour que la fonction $\varphi(d)$ atteigne son minimum en d_0 sous les contraintes (7.6).

Lemme 2 : Etant donné une matrice A_{J_0} formée en supprimant dans A_{J_0} la ligne d'indice j tel que $v_0^j < 0$, et $(I - P_{J_0})\nabla f(x^0) = 0$, le vecteur $d_0 = -(I - P_{J_0})\nabla f(x^0)$ est non nul et vérifie de plus $\langle a_j, d_0 \rangle < 0$.

Preuve : le vecteur d_0 s'écrit comme suit :

$$d_0 = -(\nabla f(x^0) + A_{J_0}^{-1}u_0), \text{ avec } u_0 = -(A_{J_0}A_{J_0}^{-1})^{-1}A_{J_0}\nabla f(x^0).$$

Si $d_0 = 0$ alors $\nabla f(x^0) + A_{J_0}^{-1}u_0 = 0$; d'autre part,

$$d_0 = -(\nabla f(x^0) + A_{J_0} u_0), \text{ avec } u_0 = -(A_{J_0} A_{J_0}^{-1})^{-1} A_{J_0} \nabla f(x^0).$$

Si $d_0 = 0$ alors $\nabla f(x^0) + A_{J_0} u_0 = 0$; d'autre part,

$$(I - P_{J_0}) \nabla f(x^0) = \nabla f(x^0) + A_{J_0} v_0 = 0. \quad (7.8)$$

Par hypothèse, on obtient après soustraction

$$A_{J_0} v_0 - A_{J_0} u_0 = u_0 a_j + \sum_{i \neq j} (v_0^i - u_0^i) a_i = 0$$

Ce qui est impossible, étant donné que $u_0^j \neq 0$ et que les vecteurs $a_i, i \in J_0$, sont linéairement indépendants.

Démontrons la seconde partie du **Lemme**.

La relation (7.8) peut s'écrire :

$$\nabla f(x^0) + \sum v_0^i a_i + (-v_0^j)(-a_j) = 0. \quad (7.9)$$

Considérons le problème :

$$\min_{\varphi} \varphi(d) = \langle \nabla f(x^0), d \rangle + 1/2 \|d\|^2,$$

sous les contraintes

$$\langle a_i, d \rangle = 0, \quad i \in J_0'$$

$$-\langle a_j, d \rangle \leq 0. \quad (7.10)$$

Du fait que $\nabla \varphi(d) = \nabla f(x^0) + d$, on a $\nabla \varphi(0) = \nabla f(x^0)$ et (7.9) est donc une condition nécessaire et suffisante pour que le point $d=0$ réalise le minimum de $\varphi(d)$ sous les contraintes (7.10).

Or, selon le lemme précédent, d_0 est solution du problème de minimisation de $\varphi(d)$ en présence des contraintes

$$A_{J_0} d = 0. \quad (7.11)$$

Supposons que $\langle a_j, d_0 \rangle \geq 0$; comme d_0 vérifie (7.11), il vérifie alors (7.10).

Mais

$$\langle \nabla f(x^0), d_0 \rangle = - \|d_0\|^2,$$

donc

$$\varphi(d_0) = \langle \nabla f(x^0), d_0 \rangle + 1/2 \|d_0\|^2 = -1/2 \|d_0\|^2 < 0,$$

ce qui contredit le fait que, dans les contraintes (7.10), la valeur minimale de $\varphi(d)$ est atteinte pour $d=0$ et vaut 0. La contradiction ainsi obtenue montre que $\langle a_i, d_0 \rangle < 0$.

Revenons à notre algorithme. On se place dans le cas 1) et on suppose que x^0 n'est pas solution du problème de programmation quadratique. On applique alors, selon l'algorithme, la méthode du gradient conjugué pour minimiser $f(x)$ sous les contraintes (7.4). Le premier pas de la méthode est dans la direction du vecteur

$$d_0 = -(I - P_{\mathcal{J}_0}) \nabla f(x^0).$$

Selon le lemme précédent, $d_0 \neq 0$, ce qui traduit le fait que le point x^0 n'est pas solution du problème (7.3)-(7.4).

On va aborder la deuxième étape, dans laquelle on démontre que $\lambda_0^* > 0$.

Lemme 3 : La valeur λ_0^* définie par

$$\lambda_0^* = \min \left\{ \frac{b_i - \langle a_i, x^0 \rangle}{\langle a_i, d_0 \rangle} ; i \in J \right\} \text{ avec } J = \{1 \leq i \leq m ; \langle a_i, d_0 \rangle > 0\},$$

est toujours positive.

Preuve : en effet, le vecteur d_0 remplit la condition (7.11) et $\langle a_i, d_0 \rangle < 0$ d'après le Lemme précédent, donc

$$\langle a_i, d_0 \rangle \leq 0, \quad i \in \tilde{J}_0 \tag{7.12}$$

D'après le choix de \tilde{J}_0 , si $i \notin \tilde{J}_0$, on a

$$\langle a_i, x^0 \rangle - b_i < 0,$$

donc

$$\lambda_0^* = \min \left\{ \frac{b_i - \langle a_i, x^0 \rangle}{\langle a_i, d_0 \rangle} ; i \in J \right\} > 0 \text{ pour } i \in J = \{1 \leq i \leq m ; \langle a_i, d_0 \rangle > 0\},$$

$$\langle a_i, x^1 \rangle - b_i = 0, i \in J_0' \text{ et } \langle a_i, x^1 \rangle - b_i < 0, i \notin J_0' \quad (7.13)$$

Et dans le second, $x^1 = x^0 + \lambda_0^* d_0$ qui représente un nouveau point initial et on teste 1) ou 2). Ceci étant, dans ce deuxième cas, x^1 vérifie, en plus des relations

$$\langle a_i, x^1 \rangle - b_i = 0, i \in J', \text{ les égalités}$$

$$\langle a_i, x^1 \rangle - b_i = 0 \text{ pour tous les } i \notin J_0', \text{ tels que}$$

$$\lambda_0^* = \frac{b_i - \langle a_i, x^0 \rangle}{\langle a_i, d_0 \rangle}$$

Ainsi $J_1 \supset J_0'$, l'inclusion étant stricte.

Reprenons le cas $\lambda^0 < \lambda_0^*$. L'algorithme du gradient conjugué continue, et tant que $\lambda^k < \lambda_k^*$, tous les points satisfont aux relations (7.13) comme x^1 .

En effet, on a $x^{k+1} = x^k + \lambda^k d_k$, et d'après (6.8) on a $d_k = (I - P_{J_0'}) d_k$, d'où

$$A_{J_0'} d_k = A_{J_0'} (I - P_{J_0'}) d_k = A_{J_0'} d_k - A_{J_0'} A_{J_0'}^{-1} (A_{J_0'} A_{J_0'}^{-1})^{-1} A_{J_0'} d_k = A_{J_0'} d_k - A_{J_0'} d_k = 0$$

i.e., en coordonnées

$$\langle a_i, d_k \rangle = 0, i \in J_0',$$

d'où

$$\langle a_i, x^{k+1} \rangle - b_i = \langle a_i, x^k \rangle - b_i + \lambda^k \langle a_i, d_k \rangle = 0, i \in J_0'.$$

D'autre part,

$$\begin{aligned} \langle a_i, x^{k+1} \rangle - b_i &= \langle a_i, x^k \rangle - b_i + \lambda^k \langle a_i, d_k \rangle < \langle a_i, x^k \rangle - b_i + \lambda_k^* \langle a_i, d_k \rangle \leq \\ &\leq \langle a_i, x^k \rangle - b_i + [\langle a_i, x^k \rangle - b_i / \langle a_i, d_k \rangle] \langle a_i, d_k \rangle = 0, \end{aligned}$$

d'où

$$\langle a_i, x^{k+1} \rangle - b_i < 0, i \notin J_0'.$$

Ainsi nous avons montré que dans le cas 1) le processus itératif génère une suite de points $x^0, x^1, \dots, x^k, k \geq 0$, et que $f(x)$ est strictement décroissante suivant cette suite, étant donné que celle-ci est formée par la méthode du gradient conjugué.

x^k étant le point terminal, alors, où bien il réalise le minimum de $f(x)$ avec les contraintes, ou bien c'est un point en lequel J_0' est strictement contenu dans J_k .

Ainsi nous avons montré que dans le cas 1) le processus itératif génère une suite de points $x^0, x^1, \dots, x^k, k \geq 0$, et que $f(x)$ est strictement décroissante suivant cette suite, étant donné que celle-ci est formée par la méthode du gradient conjugué.

x^k étant le point terminal, alors, où bien il réalise le minimum de $f(x)$ avec les contraintes, ou bien c'est un point en lequel \mathcal{J}_0 est strictement contenu dans \mathcal{J}_k .

Montrons que, avec x^0 comme point de départ, on aboutit après un nombre fini de pas à x^k , qui réalise le minimum de $f(x)$ sous les contraintes

$$\langle a_i, x \rangle - b_i = 0, i \in \mathcal{J}_k.$$

En effet, l'exposé précédent permet de dire que si la méthode du gradient conjugué ne résoud pas le problème de minimisation, cela entraînera à chaque fois une extension de l'ensemble des indices i tels que un point généré vérifie les relations $\langle a_i, x^k \rangle - b_i = 0$. Cette extension cesse évidemment au bout de n pas au plus ($n < m$, m étant le nombre de contraintes), où n est la dimension de l'espace, puisque les vecteurs $a_i, i \in \mathcal{J}_k$, sont linéairement indépendants par hypothèse. Si $m < n$ alors l'extension cesse au bout de m pas au plus, car au bout du $m^{\text{ième}}$ pas, on aura

$$\mathcal{J}_k = \{1, 2, \dots, m\}.$$

Dans le cas 2), la direction de déplacement à partir de x^0 coïncide avec le vecteur

$$d_0 = -(I - P_{\mathcal{J}_0}) \nabla f(x^0) \neq 0, \langle a_i, d_0 \rangle = 0, i \in \mathcal{J}_0.$$

On démontre donc sans peine que $\lambda_0^* > 0$ et que la méthode du gradient conjugué permet de faire un pas non nul vers le point x^1 , et en ce point $f(x^1) < f(x^0)$.

De même que précédemment, on aboutit à une suite de points $x^0, x^1, \dots, x^k, k \geq 0$, alors, ou bien $\mathcal{J}_k \supset \mathcal{J}_0$, ou bien x^k réalise le minimum de $f(x)$ sous les contraintes (7.5).

Tous les ensembles \mathcal{J}_k font partie de l'ensemble $\mathcal{J} = \{1, 2, \dots, m\}$ et ils sont donc en nombre fini, Il s'ensuit alors l'arrêt du processus proposé après un nombre fini de pas. Or, l'arrêt ne peut avoir lieu que si l'on a trouvé le minimum de $f(x)$ sous les contraintes (1.2).

IV.2.8. SCHEMA DE L'ITERATION :

Soient x^0 un point quelconque tel que $Ax^0 \leq b$, $J_0 = \{1 \leq i \leq m ; \langle a_i, x^0 \rangle - b_i = 0\}$ et A_{J_0} la matrice dont les colonnes sont les vecteurs a_i , $i \in J_0$.

Posons $P_{J_0} = A_{J_0}^{-1}(A_{J_0} A_{J_0}^{-1})^{-1} A_{J_0}$ et $v_0 = -(A_{J_0} A_{J_0}^{-1})^{-1} A_{J_0} \nabla f(x^0)$.

1) Si $(I - P_{J_0}) \nabla f(x^0) = 0$ alors on passe à 2),

Si $(I - P_{J_0}) \nabla f(x^0) \neq 0$ alors on passe à 3),

2) Si $v_0^i \geq 0$, $i \in J_0$ alors x^0 est solution du problème (1.1)-(1.2).

Sinon, i.e. il existe $j \in J_0$ tel que $v_0^j < 0$, on pose alors $J_0 = J_0 - \{j\}$ et on passe à 3).

3) Procédure du gradient conjugué pour la minimisation de

$$f(x) = 1/2 \langle x, Cx \rangle + \langle x, d \rangle \quad (**)$$

sous les contraintes

$$\langle a_i, x \rangle - b_i = 0 \quad i \in J_0.$$

a) On prend x^0 comme point de départ, et on pose $d_0 = -(I - P_{J_0}) \nabla f(x^0)$

b) A l'itération k , on a x^k ; pour déterminer x^{k+1} on procède de la façon suivante :

on définit

$$d_k = -(I - P_{J_0}) \nabla f(x^k) + \beta^k d_{k-1},$$

avec

$$\beta^k = \frac{\|(I - P) \nabla f(x^{k+1})\|^2}{\|(I - P) \nabla f(x^k)\|^2}$$

$$\lambda^k = \frac{\|\nabla f(x^k)\|^2}{\langle d_k, C d_k \rangle} \quad k=0, 1, 2, \dots$$

et

$$\lambda_k^* = \min \left\{ \frac{b_i - \langle a_i, x^k \rangle}{\langle a_i, d_k \rangle} ; i \in J \right\} \quad \text{où } J = \{i ; \langle a_i, d_k \rangle > 0\}.$$

Si $\lambda^k < \lambda_k^*$ alors $x^{k+1} = x^k + \lambda^k d_k$, faire $k \leftarrow k+1$ et retourner à b) jusqu'à l'obtention d'un point qui réalise le minimum du problème (**), qu'on notera x^* et on passe à 4).

Si $\lambda^k \geq \lambda_k^*$ alors $x^{k+1} = x^k + \lambda_k^* d_k$; on pose $x^* = x^{k+1}$ et on passe à 4).

4) on pose $x^0 = x^*$ et $\mathcal{J}_0 = \{1 \leq i \leq m, \langle a_i, x^* \rangle - b_i \leq 0\}$ et on recommence le processus (passage à 1)).

IV.3. DUALITE ET FONCTION DE LAGRANGE :

Soit le problème

$$\min f(x), \tag{8.1}$$

sous les contraintes

$$g_i(x) \leq 0 \text{ pour } 1 \leq i \leq m. \tag{8.2}$$

La fonction de Lagrange associée au problème (8.1)-(8.2) est définie par :

$$L(x, u) = \begin{cases} f(x) + \langle u, g(x) \rangle & \text{si } u \geq 0 \\ -\infty & \text{sinon,} \end{cases}$$

où $u = (u_1, u_2, \dots, u_m)^t$ et $g(x) = (g_1(x), g_2(x), \dots, g_m(x))^t$.

Le vecteur $u = (u_1, u_2, \dots, u_m)^t$ est le vecteur de *la variable duale*. Cette fonction joue un rôle fondamental dans la théorie de la dualité.

On définit la fonction primale par

$$L_p(x) = \text{Sup}\{L(x, u) ; u \geq 0\},$$

et la fonction duale par

$$L_d(u) = \text{Inf}\{L(x, u) ; x \in \mathbb{R}^n\}.$$

La fonction primale n'est autre que $f(x)$. Comme la fonction primale (resp duale) est une fonction de x (resp de u), on peut formuler les deux problèmes d'optimisation suivants :

Problème primal :

$$\min\{L_p(x) ; g_i(x) \leq 0 ; 1 \leq i \leq m\} = \min\{f(x) ; g_i(x) \leq 0 ; 1 \leq i \leq m\}. \quad (8.3)$$

Problème dual :

$$\max\{L_d(u) ; u \geq 0\}. \quad (8.4)$$

On définit aussi un point-selle de la *fonction de Lagrange* (x^*, u^*) , $u^* \geq 0$, par :

$$L(x^*, u) \leq L(x^*, u^*) \leq L(x, u^*) \quad \forall u \geq 0 \text{ et } \forall x \in \mathbb{R}^n. \quad (8.5)$$

On notera D l'ensemble des solutions réalisables de (8.1)-(8.2), c'est à dire :

$$D = \{x \in \mathbb{R}^n ; g_i(x) \leq 0, 1 \leq i \leq m\}. \quad (8.6)$$

Pour que le domaine D défini par les contraintes $g_i(x) \leq 0$ ($1 \leq i \leq m$) vérifie en $x^0 \in D$ l'hypothèse de qualification des contraintes ou l'hypothèse de SLATER (1950), il suffit que la condition suivante soit réalisée :

toutes les fonctions g_i sont convexes et D a un intérieur non vide.

On a le théorème de KUHN & TUCKER (1951) qui donne une condition nécessaire et suffisante pour qu'un point soit le minimum d'un problème de programmation non-linéaire :

THEOREME 1 [10] : *On suppose que les fonctions f et g_i ($1 \leq i \leq m$) sont continûment différentiables, et que l'hypothèse de qualification des contraintes est vérifiée en $x^0 \in D$, où D est défini par (8.6). Alors le point x^* est solution du problème primal si et seulement si, il existe $u^* \geq 0$ tel que (x^*, u^*) constitue un point-selle de la fonction de Lagrange.*

Dans ce cas u^* est solution du problème dual. Autrement dit, (x^*, u^*) est un point-selle de L défini par (8.5), si et seulement si

$$\nabla_x L(x^*, u^*) = 0 \quad (8.7)$$

$$\nabla_u L(x^*, u^*) \leq 0 \quad (8.8)$$

$$\langle u, \nabla_u L(x^*, u^*) \rangle = 0 \quad (8.9)$$

$$u^* \geq 0 \quad (8.10)$$

Où ∇_x, ∇_u représentent respectivement les gradients par rapport à x et u.

La théorie de la dualité nous permet de formuler et de résoudre le problème dual au lieu du problème primal, ce qui est dans certains cas, plus avantageux et plus commode.

IV.3.1. LE DUAL DU PROBLEME QUADRATIQUE (1.1)-(1.2) :

Reprenons le problème de la minimisation de la fonction quadratique

$$f(x) = 1/2 \langle x, Cx \rangle + \langle x, d \rangle,$$

sous les contraintes

$$Ax \leq b.$$

Selon l'étude faite précédemment, le dual de ce problème est :

$$\max\{\min\{1/2 \langle x, Cx \rangle + \langle x, d \rangle + \langle u, Ax - b \rangle ; x \in \mathbb{R}^n\} ; u \geq 0\}. \quad (8.11)$$

Le problème sans contraintes

$$\min\{1/2 \langle x, Cx \rangle + \langle x, d \rangle + \langle u, Ax - b \rangle ; x \in \mathbb{R}^n\},$$

a pour solution

$$x^* = -C^{-1}[d + A'u]. \quad (8.12)$$

Substituons (8.12) dans (8.11). On aura :

$$\min\{1/2 \langle u, AC^{-1}A'u \rangle + \langle u, b + AC^{-1}d \rangle ; u \geq 0\}. \quad (8.13)$$

Ce problème dual est lui aussi un programme quadratique, mais le nombre de variables est m au lieu de n ($m < n$), ce qui est avantageux. On remarque d'autre part que le problème dual possède des contraintes très simples de positivité.

Il est parfois recommandé, compte-tenu des avantages cités ci-dessus, de résoudre le problème dual au lieu du problème primal, ce qui nous a conduit à décrire dans le paragraphe qui suit la méthode de résolution du problème dual (8.13).

IV.3.2. METHODE DE RESOLUTION DU PROBLEME DUAL (8.12) :

Le problème dual est : minimiser

$$F(u) = 1/2 \langle u, AC^{-1}A'u \rangle + \langle u, b + AC^{-1}d \rangle ; u \geq 0, \quad (9.1)$$

sous les contraintes

$$u_i \geq 0 \quad \text{pour } i=1, \dots, m, \quad (9.2)$$

avec $u = (u_1, u_2, \dots, u_m)^t \in \mathbb{R}^n$.

THEOREME 1 [15] : Pour que le point u^* réalise le minimum de la fonction $F(\cdot)$ sur l'ensemble

$$u_i \geq 0, \quad i=1, \dots, m,$$

il faut et il suffit qu'on ait les relations

$$\frac{\partial F(u^*)}{\partial u_j} \geq 0 \quad \text{si } u_j = 0 \quad \text{pour } j=1, 2, \dots, m,$$

et

$$\frac{\partial F(u^*)}{\partial u_j} = 0 \quad \text{si } u_j \neq 0 \quad \text{pour } j=1, 2, \dots, m.$$

Pour résoudre (9.1)-(9.2), on va décrire l'algorithme suivant, lequel est basé sur celui présenté dans le paragraphe IV.2.7, mais qui se simplifie considérablement. Soit u^0 un point arbitraire vérifiant $u_i^0 \geq 0$ pour $i=1, \dots, m$; posons

$$\mathcal{J}(u) = \{1 \leq i \leq m ; u_i = 0\}.$$

Décrivons les opérations à partir de u^0 , et calculons l'ensemble $\mathcal{J}(u^0)$; deux cas peuvent se présenter :

1) $[\nabla F(u^0)]^i = 0, i \notin \mathcal{J}(u^0)$, où $[\nabla F(u^0)]^i$ est la $i^{\text{ème}}$ composante du vecteur $\nabla F(u^0)$.

Dans ce cas, u^0 réalise le minimum de $F(u)$ sous les contraintes $u_i = 0, i \in \mathcal{J}(u^0)$. Si de plus $[\nabla F(u^0)]^i > 0, i \in \mathcal{J}(u^0)$, alors u^0 est solution du problème, car en ce point les conditions nécessaires et suffisantes de minimum sont vérifiées.

Soit maintenant, $[\nabla F(u^0)]^i < 0$ pour certains $i \in \mathcal{J}(u^0)$.

Posons $\mathcal{J}' = \{i \in \mathcal{J}(u^0) : [\nabla F(u^0)]^i \geq 0\}$. On résoud le problème

$$\min F(u),$$

sous $u_i = 0 \quad i \in \mathcal{J}'$.

Par la méthode du *gradient conjugué*, on doit toujours calculer λ_k^* :

$$\lambda_k^* = \min \left\{ -\frac{u_i^k}{d_k^i} ; i \in J \right\} \text{ avec } J = \{i \in \mathcal{J}' : d_k^i < 0\},$$

et on compare λ^k et λ_k^* .

Si $\lambda^k < \lambda_k^*$ alors $u_i^{k+1} = u_i^k + \lambda^k d_k^i$, $i \in \mathcal{J}'$ et $u_i^{k+1} = u_i^k = 0$, $i \in \mathcal{J}'$.

Si $\lambda^k \geq \lambda_k^*$ alors $u_i^{k+1} = u_i^k + \lambda_k^* d_k^i$, $i \in \mathcal{J}'$ et $u_i^{k+1} = u_i^k = 0$, $i \in \mathcal{J}'$.

Le processus s'arrête après un nombre fini de pas et on trouve un point u^{k+1} tel que, soit $\lambda^k \geq \lambda_k^*$, soit $F(u)$ atteint en ce point son minimum sous les conditions $u_i = 0 \quad i \in \mathcal{J}'$. Dans chacun des cas le processus recommence avec u^{k+1} au départ.

II) Il existe des indices i tels que $[\nabla F(u^0)]^i \neq 0$, $i \in \mathcal{J}(u^0) = \mathcal{J}_0$; on minimise alors

$$F(u),$$

sous $u_i = 0 \quad i \in \mathcal{J}_0$,

par la méthode du *gradient conjugué*. De plus, on calcule à chaque pas (comme pour I)) la quantité

$$\lambda_k^* = \min \left\{ -\frac{u_i^k}{d_k^i} ; i \in J \right\} \text{ avec } J = \{i \in \mathcal{J}_0 : d_k^i < 0\}$$

et on compare λ^k et λ_k^* .

Si $\lambda^k < \lambda_k^*$ alors $u_i^{k+1} = u_i^k + \lambda^k d_k^i$, $i \in \mathcal{J}_0$ et $u_i^{k+1} = u_i^k = 0$, $i \in \mathcal{J}_0$.

Si $\lambda^k \geq \lambda_k^*$ alors $u_i^{k+1} = u_i^k + \lambda_k^* d_k^i$, $i \in \mathcal{J}_0$ et $u_i^{k+1} = u_i^k = 0$, $i \in \mathcal{J}_0$.

Le processus se termine comme dans le cas 1).

IV.3.3. SCHEMA DE L'ITERATION :

Soient u^0 un point tel que $u_i^0 \geq 0$ pour $1 \leq i \leq m$ et $\mathcal{J}_0 = \{1 \leq i \leq m ; u_i^0 = 0\}$.

1) Si $[\nabla f(u^0)]^i = 0$, $i \notin \mathcal{J}_0$ on passe à 2).

Si $[\nabla f(u^0)]^i \neq 0$, pour un certain $i \in \mathcal{J}_0$, on passe à 3).

2) Si $[\nabla f(u^0)]^i \geq 0$, $i \in \mathcal{J}_0$, alors u^0 est solution du problème (9.1)-(9.2).

Sinon, i.e. il existe $j \in \mathcal{J}_0$ tel que $[\nabla f(u^0)]^j < 0$; on pose alors

$\mathcal{J}_0 = \mathcal{J}_0 - \{i \in \mathcal{J}_0 : [\nabla f(u^0)]^i < 0\}$ et on passe à 3).

3) Procédure du gradient conjugué pour la résolution de la minimisation

$$f(u) = 1/2 \langle u, AC^{-1}A^t \rangle + \langle u, AC^{-1}d \rangle \quad (**),$$

sous les contraintes

$$u_i = 0 ; i \in \mathcal{J}_0.$$

a) On prend comme point de départ u^0 et on pose $d_0 = -\nabla F(u^0)$.

b) A l'itération k , on a x^k ; pour déterminer x^{k+1} on procède de la manière suivante :

on définit

$$d_k = -\nabla F(u^k) + \beta_{k-1} d_{k-1},$$

avec

$$\beta_{k-1} = \frac{\|\nabla F(u^k)\|^2}{\|\nabla F(u^{k-1})\|^2}$$

On définit le pas de déplacement par

$$\lambda_k = \frac{\langle \nabla F(u^k), d_k \rangle}{\langle d_k, Cd_k \rangle}$$

et

$$\lambda_k^* = \min \left\{ -\frac{u_i^k}{d_k^i} ; i \in J \right\} \quad \text{avec } J = \{i \notin \mathcal{J}_0 : d_k^i < 0\}$$

Si $\lambda^k < \lambda_k^*$ alors $u_i^{k+1} = u_i^k + \lambda^k d_k^i$, $i \notin \mathcal{J}_0$ et $u_i^{k+1} = u_i^k = 0$, $i \in \mathcal{J}_0$.

Faire $k \rightarrow k+1$ et retourner à b), jusqu'à obtention d'un point, noté u^* , qui réalise le minimum du problème (***) et on passe à 4).

Si $\lambda^k \geq \lambda_k^*$ alors $u_i^{k+1} = u_i^k + \lambda_k^* d_k^i$, $i \notin \mathcal{J}_0$ et $u_i^{k+1} = u_i^k = 0$, $i \in \mathcal{J}_0$.

On pose $u^* = u^{k+1}$ et on passe à 4).

4) On pose $u^0 = u^*$ et $\mathcal{J}_0 = \{1 \leq i \leq m ; u^* = 0\}$, puis on recommence le processus (passage à 1)).

IV.3.4. ASPECT NUMERIQUE :

L'algorithme du *gradient conjugué conditionnel*, décrit dans le paragraphe IV.2.7., comprend en fait une seule itération compliquée, à savoir la projection du gradient sur un sous espace, i.e. le calcul de

$$(I - P)\nabla f(x) = \nabla f(x) - P.\nabla f(x)$$

La première façon d'agir consiste à calculer directement la matrice $P_{\mathcal{J}}$, i.e. $P_{\mathcal{J}} = A_{\mathcal{J}}^t (A_{\mathcal{J}} A_{\mathcal{J}}^t)^{-1} A_{\mathcal{J}}$ et dans ce cas on doit calculer sans cesse $(A_{\mathcal{J}} A_{\mathcal{J}}^t)^{-1}$.

Cette façon de faire produit une accumulation éventuelle d'une grande erreur de calcul. Pour remédier à ce phénomène, on procède comme suit :

en posant $v^k = -(A_{\mathcal{J}} A_{\mathcal{J}}^t)^{-1} A_{\mathcal{J}} \nabla f(x^k)$, le calcul à chaque itération de la quantité $(I - P_{\mathcal{J}})\nabla f(x^k)$ se ramène à celui de l'expression :

$$\nabla f(x^k) + A_{\mathcal{J}}^t v^k \tag{10.1}$$

Dans (10.1), la seule opération compliquée à effectuer est le calcul du vecteur v^k . Or, calculer v^k revient à résoudre le système suivant

$$(A_{\mathcal{J}} A_{\mathcal{J}}^t) v^k = -A_{\mathcal{J}} \nabla f(x^k) . \tag{10.2}$$

Le vecteur $(I - P_{\mathcal{J}})\nabla f(x^k)$ est obtenu facilement par la formule

$$\begin{aligned} (I - P_{\mathcal{J}})\nabla f(x^k) &= \nabla f(x^k) - P_{\mathcal{J}}\nabla f(x^k) = \nabla f(x^k) - A_{\mathcal{J}}^t (A_{\mathcal{J}} A_{\mathcal{J}}^t)^{-1} A_{\mathcal{J}} \nabla f(x^k) = \\ &= \nabla f(x^k) + A_{\mathcal{J}}^t v^k \end{aligned}$$

Ainsi, en procédant de cette façon, le problème se ramène à une répétition d'applications d'une procédure standard pour la résolution du système (10.2).

C'est justement par le fait de passer au problème dual, qu'on évite la résolution, à chaque itération, d'un système d'équations, ce qui permet d'avoir une bonne convergence de l'algorithme du *gradient conjugué*.

IV.4. APPLICATIONS :

IV.4.1. PROBLEME DE LA REGRESSION ISOTONE :

Le problème de la régression isotone consiste à minimiser

$$\sum_{i=1}^n w_i (g_i - x_i)^2, \tag{11.1}$$

sous les contraintes

$$x_i - x_{i+1} \leq 0 \text{ pour } i=2, \dots, n, \tag{11.2}$$

avec $w_i > 0$, $1 \leq i \leq n$, $g = (g_1, g_2, \dots, g_n)^t \in \mathbb{R}^n$, $x = (x_1, x_2, \dots, x_n)^t \in \mathbb{R}^n$.

Posons

$$W = \begin{bmatrix} w_1 & 0 & \dots & \dots & 0 \\ 0 & w_2 & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \dots & w_n \end{bmatrix} \quad \text{et} \quad A = \begin{bmatrix} 1 & -1 & 0 & \dots & \dots & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \dots & 1 & -1 \end{bmatrix}$$

La matrice W est une nxn-matrice diagonale, symétrique, définie positive, et dont les éléments diagonaux (w_i , $1 \leq i \leq n$) sont non-négatifs, tandis que la matrice A est une n-1xn-matrice, représentant les contraintes, dont la ligne i est composée d'éléments nuls sauf les $i^{\text{ème}}$ et $(i+1)^{\text{ème}}$ composantes qui valent 1 et -1, respectivement. Ainsi, le problème (11.1)-(11.2) est équivalent à

$$\min 1/2 \langle x, Wx \rangle + \langle x, -Wg \rangle, \tag{11.3}$$

sous les contraintes

$$Ax \leq 0. \tag{11.4}$$

On voit facilement que le problème de *la régression isotone* se ramène à un problème de programmation quadratique avec des contraintes linéaires. De plus, on remarque que les lignes de la matrice A sont linéairement indépendantes.

On se propose alors de résoudre le problème (11.3)-(11.4) par la méthode du *gradient conjugué conditionnel* décrite dans le paragraphe IV.2.7.

Selon l'étude faite au paragraphe IV.3., le problème dual du problème de la régression isotone s'écrit ainsi :

$$\min 1/2 \langle u, AW^{-1}A^t u \rangle + \langle u, -Ag \rangle, \tag{11.5}$$

sous les contraintes

$$u_i \geq 0 \text{ pour } i=1, \dots, n-2. \tag{11.6}$$

La matrice qui définit le terme quadratique est donnée par

$$AW^{-1}A^t = \begin{bmatrix} \frac{1}{w_1} + \frac{1}{w_2} & -\frac{1}{w_2} & \dots & \dots & \dots & 0 \\ -\frac{1}{w_2} & \frac{1}{w_2} + \frac{1}{w_3} & -\frac{1}{w_3} & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \dots & -\frac{1}{w_{n-1}} \\ 0 & 0 & 0 & -\frac{1}{w_{n-1}} & \frac{1}{w_{n-1}} + \frac{1}{w_n} & \dots \end{bmatrix}$$

On constate que la matrice qui définit le terme quadratique du problème dual est d'un type simple, puisqu'elle est tridiagonale .

La forme générale des lignes de la matrice $AW^{-1}A^t$ est la suivante :

$$(AW^{-1}A^t)_1 = \left(\frac{1}{w_1} + \frac{1}{w_2}, -\frac{1}{w_2}, 0, 0, \dots, 0 \right)$$

$$(AW^{-1}A^t)_i = \left(0, 0, \dots, -\frac{1}{w_i}, \frac{1}{w_i} + \frac{1}{w_{i+1}}, -\frac{1}{w_{i+1}}, 0, 0, \dots, 0 \right) \text{ pour } i=2, \dots, n-2.$$

avec

$$(AW^{-1}A^t)_{i(i-1)} = -\frac{1}{w_i}$$

$$(AW^{-1}A^t)_{i(i)} = \frac{1}{w_i} + \frac{1}{w_{i+1}}$$

$$(AW^{-1}A^t)_{i(i+1)} = -\frac{1}{w_{i+1}}$$

et $(AW^{-1}A^t)_{i(j)} = 0$ pour $j \neq i-1, i, i+1$.

$$(AW^{-1}A^t)_{n-1} = (0, 0, \dots, 0, -\frac{1}{w_{n-1}}, \frac{1}{w_{n-1}} + \frac{1}{w_n})$$

Une fois u^* déterminé, on déduit immédiatement la solution du problème de la régression isotone :

$$x^* = -W^{-1}A^t u^* + W^{-1}g.$$

Tout calcul fait, la valeur explicite de x^* est égale à :

$$x^* = \left(\frac{g_1 - u_1^*}{w_1}, \frac{u_1^*}{w_2} + \frac{g_2 - u_2^*}{w_2}, \dots, \frac{u_{n-2}^*}{w_{n-1}} + \frac{g_{n-1} - u_{n-1}^*}{w_{n-1}}, \frac{g_n + u_n^*}{w_n} \right)$$

Donc, pour la résolution du problème de la régression isotone, il est préférable de passer au problème dual, qui présente des avantages sur le problème primal. Ainsi, la méthode du gradient conjugué appliquée au problème dual semble *simple et commode*.

IV.4.2. REGRESSION CONCAVE :

Le problème de la régression concave s'écrit ainsi :

$$\min \sum_{i=1}^n w_i (g_i - x_i)^2, \tag{11.7}$$

sous les contraintes

$$\frac{x_i - x_{i-1}}{y_i - y_{i-1}} \leq \frac{x_{i+1} - x_i}{y_{i+1} - y_i} \text{ pour } i = 2, \dots, n-1, \tag{11.8}$$

avec $w_i > 0, 1 \leq i \leq n, g = (g_1, g_2, \dots, g_n)^t \in \mathbb{R}^n, x = (x_1, x_2, \dots, x_n)^t \in \mathbb{R}^n$.

Les variables $y_i, 1 \leq i \leq n$, vérifient la relation $0 < y_1 < y_2 < \dots < y_n$.

Posons

$$A = \begin{bmatrix} y_3 - y_2 & y_3 - y_1 & y_2 - y_1 & 0 & 0 & 0 & \dots & 0 \\ 0 & y_4 - y_3 & y_4 - y_2 & y_3 - y_2 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & y_n - y_{n-1} & y_n - y_{n-2} & \dots & y_{n-1} - y_{n-2} \end{bmatrix}$$

Comme dans le cas 11.1) on a l'équivalence entre le problème de la regression concave et le problème de programmation quadratique suivant :

$$\min 1/2 \langle x, Wx \rangle + \langle x, -Wg \rangle, \tag{11.9}$$

sous les contraintes

$$Ax \leq 0. \tag{11.10}$$

La matrice A est donnée par l'expression ci-dessus.

REFERENCES

- 1) **A.AUSLENDER(1976)**
Optimisation : méthodes numériques.
Masson, Paris.
- 2) **J.CEA(1971)**
Optimisation : théorie et algorithmes.
Dunod, Paris.
- 3) **L.COLLATZ and W.WETTERLING(1975)**
Optimization Problems.
Springer-verlag New York Heidelberg Berlin.
- 4) **I.EKELAND ET R.TEMAM(1974)**
Analyse Convexe et Problèmes Variationnels.
Dunod, Paris.
- 5) **N.GASTINEL(1966)**
Analyse numérique linéaire.
Hermann, Paris.
- 6) **P.E.GILL, W.MURRAY and M.H.WRIGHT(1981)**
Practical Optimization.
Academic Press London New York Toronto.
- 7) **C.HILDRETH(1957)**
A Quadratic Programming Procedure.
Naval Res Logist Quart, 4, pp.79-85.
- 8) **L.S.JACOBY, J.S.KOWALIK and J.T.PIZZO(1972)**
Iterative Methods for Nonlinear Optimization Problems.
Prentice-Hall, Inc, New-Jersey.
- 9) **P.JOLY(1986)**
Présentation de synthèse des méthodes de gradient conjugué.
Mathematical Modelling and Numerical Analysis, vol 20, 4, pp. 639-665.
- 10) **P.J.LAURENT(1972)**
Approximation et optimisation.
Hermann, Paris.
- 11) **A.LENT and Y.CENSOR(1980)**
Extensions of Hildreth's Row-Action Method for Quadratic Programming.
SIAM J.Control and Optimization, vol 18, 4, pp.444-454.

12) **M.MINOUX(1983)**

Programmation mathématique : théorie et algorithmes.
Dunod, Paris.

13) **B.PCHENITCHNY et Y.DANILINE(1977)**

Methodes numériques dans les problèmes d'extrémum.
Edition Mir.

14) **R.T.ROCKAFELLAR(1972)**

Convex Analysis.
Princeton University Press, Princeton.

15) **G.ZOUTENDIJK(1966)**

Nonlinear Programming : a numerical survey.
SIAM J.on Control 4, 1, pp 194-210.

CHAPITRE V

***PRESENTATION DE SYNTHESE DES METHODES DE
PROJECTIONS.***



V.1. INTRODUCTION :

La recherche d'un point commun à une famille de convexes fermés se formule de la manière suivante:

(P) " Trouver un point x appartenant à $\cap C_i$, pour $i \in I$ "

Où les C_i sont des convexes fermés et I un ensemble d'indices de cardinal quelconque. Ce problème important, permet de résoudre des programmes linéaires, à l'aide de la théorie de la dualité, des systèmes d'égalités et d'inégalités linéaires.

La plupart des méthodes existantes pour résoudre ce problème sont fondées sur les propriétés de projecteurs. L'ensemble de ces méthodes présentées repose sur la possibilité d'effectuer facilement des projections sur chacun des convexes initiaux. Les plus connues de ces méthodes sont les méthodes de relaxation dues à S.KACZMARZ[12], G.CIMMINO[12].

On peut regrouper les méthodes proposées en deux grandes classes. La classe des méthodes barycentriques dont le processus itératif s'inspire du procédé de G.CIMMINO(N.GASTINEL[12]), généralisé au cas des convexes quelconques par A.AUSLENDER[3], G.PIERRA[21], Y.CENSOR[8] et A.De PIERRO[10], et la classe des méthodes de projections successives, dues à L.G. GUBIN et al[14] qui généralise l'approche de S.KACZMARZ(N.GASTINEL[12]) qui consiste à résoudre un système linéaire de grande taille et la méthode de relaxation de S.AGMON, T.S.MOTZKIN et I.J.SCHÖENBERG[1],[20] qui consiste à chercher un point de l'intersection d'un ensemble fini de demi-espaces.

Notre travail présente tout d'abord une synthèse méthodologique des algorithmes de projections, montre d'autre part que ces algorithmes se déduisent en général de la méthode classique du sous-gradient utilisé pour minimiser une fonction convexe sans contraintes.

Dans un premier temps, on étudie le problème posé par la recherche de l'intersection de deux convexes ou par le calcul de leur distance la distance dans le cas où l'intersection est vide. On montre dans la première partie que ce problème est équivalent à un problème d'optimisation convexe avec contraintes que nous résolvons par la méthode de gradient projeté proposée en 1960 par J.P.ROSEN[27].

On constate que la méthode des projections successive étudiée dans la Thèse de G.PIERRA[21] n'est qu'un cas particulier de la méthode précédente. Dans le cas où l'un des deux convexes est une variété linéaire, on a pu donner le *pas optimal* de déplacement de cette méthode. On élabore, dans la deuxième partie, une méthode nouvelle de recherche d'un point appartenant à deux convexes, ou le calcul de leur distance dans

le cas où l'intersection est vide, moyennant la représentation de ce problème dans l'espace produit $\mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n$.

Dans un deuxième lieu, on s'intéresse au problème de recherche d'un point commun à une famille de convexes fermés.

Dans la troisième partie, on représente le problème de recherche d'un point commun à une famille de convexes fermés dans l'espace produit, on montre qu'on est en présence d'un problème de recherche de l'intersection de deux convexes -ou le calcul de leur distance dans le cas où l'intersection est vide- dans l'espace produit, dont la résolution suit les voies étudiées dans la première partie. On applique à ce problème la méthode de J.P.ROSEN. On montre alors en revenant aux espaces facteurs, que l'on obtient le processus itératif de recherche d'un point de l'intersection des convexes.

La quatrième partie généralise l'étude faite à la deuxième partie, au cas de plusieurs convexes. La méthode présentée consiste à représenter l'intersection des convexes dans un espace produit. On obtient ainsi un problème de minimisation d'une fonction différentiable sur l'espace produit qu'on propose de résoudre par la méthode de décomposition due à A.AUSLENDER et B.MARTINET[3]. On constate que cette méthode est équivalente à la méthode classique du gradient pour la minimisation d'une fonction sur l'espace \mathbb{R}^n . Le processus itératif produit par cette méthode est le même que celui donné par la méthode présentée à la troisième partie.

Le propos de la cinquième partie est de représenter le problème (P) sous forme d'un problème de minimisation d'une fonction non différentiable sur l'espace \mathbb{R}^n , qu'on résoud par la méthode du sous-gradient qu'on note sous-gradient(1). On constate que le processus itératif produit par cet algorithme n'est autre que la méthode de relaxation de L.G.GUBIN et al[14], La représentation du problème (P) dans l'espace produit permet de retrouver le même processus itératif des méthodes exposées dans la partie 3 et la partie 4 par application de l'étude précédente. On intitule cette approche la méthode du sous gradient(2).

La sixième partie est consacrée à l'étude dual du problème précédent dans le cadre de la théorie du minimax. On présente une méthode pour résoudre le problème dual qui est plus simple que le primal, et l'on remarque que c'est un cas particulier de la méthode du sous-gradient (1).

L'ensemble des méthodes présentées repose sur la possibilité d'effectuer facilement des projections sur chacun des convexes initiaux. Le cas où les projections sont difficiles à obtenir est étudié, par croisement d'algorithmes, dans la huitième partie.

Pour terminer on expose les méthodes de projection dans un cadre théorique général en donnant l'aspect géométrique de ces méthodes chaque fois que cela est possible, ainsi que les théorèmes de convergence.

La fin de cet exposé fait le lien entre ces méthodes et les méthodes d'optimisation citées ci-dessus. On remarque que toutes les méthodes de type barycentrique font partie de la classe de méthodes connues sous le nom de méthodes du sous-gradient.

On reproche toujours aux méthodes de projections de converger lentement et cela s'explique par le fait qu'elles se déduisent des méthodes du sous-gradient qui ont cet inconvénient.

V.2. APPLICATION DE LA METHODE DU GRADIENT PROJETE A LA RECHERCHE DE L'INTERSECTION DE (OU DE LA DE DISTANCE ENTRE) DEUX CONVEXES FERMES :

Le problème de la recherche d'un point appartenant à l'intersection de deux convexes (ou dans le cas où cette intersection est vide, de la distance de deux convexes et de deux points réalisant cette distance) se rencontre dans un certain nombre de domaines ([18], [23]).

La méthode généralement proposée pour résoudre ce problème est celle des projections successives ([4], [14], [21]) qui consiste à projeter un point sur le premier convexe, puis à projeter cette projection sur le second convexe et ainsi de suite.

Cette partie propose une méthode générale fondée sur la méthode du gradient projeté dont l'algorithme de projections successives n'est qu'un cas particulier. Lorsque l'un des deux convexes est une variété linéaire on donne de plus une stratégie optimale de choix du pas.

V.2.1. POSITION DU PROBLEME :

Soient l'espace \mathbb{R}^n muni du produit scalaire usuel (\cdot, \cdot), de la norme euclidienne $\|\cdot\|$ et de la distance associée notée $d(\cdot, \cdot)$. C_i $i=1,2$ deux convexes fermés de \mathbb{R}^n . On suppose de plus connus les projecteurs respectifs sur chaque convexes. Posons $C=C_1 \times C_2$ et considérons le problème :

$$\text{Trouver } (x^*, y^*) \in C \text{ tel que } 1/2 \|x^* - y^*\|^2 = \min\{1/2 \|x - y\|^2 ; (x, y) \in C\} \quad (P_1)$$

Qui est équivalent aux problèmes suivants :

$$\text{Trouver } x^* \in C_1 \text{ tel que } 1/2 \|x^* - P_{C_2}(x^*)\|^2 = \min\{1/2 \|x - P_{C_2}(x)\|^2 ; x \in C_1\} \quad (P_2)$$

$$\text{Trouver } y^* \in C_2 \text{ tel que } 1/2 \|y^* - P_{C_1}(y^*)\|^2 = \min\{1/2 \|y - P_{C_1}(y)\|^2 ; y \in C_2\} \quad (P_3)$$

REMARQUE : Le cadre d'application des méthodes présentées suppose que les projecteurs P_{C_i} sont simples à réévaluer numériquement. C'est par exemple le cas de convexes tels que des boules fermées euclidiennes, des sous-espaces affines ou des demi-espaces fermés.

V.2.2. CARACTERISATION DE L'ENSEMBLE DE SOLUTION DU PROBLEME (P₁) :

L'équivalence des 3 propositions suivantes est démontrée Dans [21],

1) $(x^*, y^*) \in C_1 \times C_2 : d(x^*, y^*) = d(C_1, C_2)$.

2) $(x^*, y^*) \in C_1 \times C_2 : d(x^*, y^*) = d(x^*, C_2) = d(C_1, y^*)$.

3) $x^* = P_{C_1}(y^*)$ et $y^* = P_{C_2}(x^*)$.

REMARQUE : La définition de la distance entre deux convexes conduit à noter on notera l'ensemble de solution du problème (P₁) par :

$$\mathcal{P} = \{(x^*, y^*) \in C ; d(x^*, y^*) = d(C_1, C_2)\}$$

C'est à dire, dans le cas où $C_1 \cap C_2 \neq \emptyset$:

$$\mathcal{P} = \{(x^*, x^*) \in \mathbb{R}^n \times \mathbb{R}^n ; x^* \in C_1 \cap C_2\}$$

il convient donc, pour chercher la distance ou l'intersection de deux convexes de résoudre le problème d'optimisation convexe avec contraintes suivant :

$$\min \{1/2 \|x - P_{C_2}(x)\|^2 ; x \in C_1\}. \quad (P_2)$$

Nous résolvons ce problème par la méthode du gradient projeté qui est particulièrement intéressante dans le cas de contraintes linéaires. Elle consiste à chaque étape de projeter le déplacement sur le convexe C₁, afin de s'assurer que ce nouveau point obtenu appartienne à l'ensemble des solutions réalisables.

V.2.3. PRESENTATION DE LA METHODE DU GRADIENT PROJETE :

L'algorithme du gradient projeté pour la résolution du problème (P₂) est conforme à la procédure suivante :

(a) A l'itération k=0 on est en $x^0 \in C_1$.

(b) A l'itération courante k on est en x^k .

calculer $\nabla f(x^k)$ puis $z^k = P_{C_1}[x^k - \nabla f(x^k)]$

Définir le déplacement $d^k = z^k - x^k$

(c) Si $d^k \neq 0$ déterminer x^{k+1} par :

$$x^{k+1} = x^k + \lambda^k d^k \quad \lambda^k > 0 \text{ pour tout } k > 0$$

Faire $k \leftarrow k+1$ et retourner à (b).

En résumé, le processus itératif de l'algorithme de gradient projeté est décrit, à partir d'un point arbitraire $x^0 \in C_1$ par l'équation :

$$x^{k+1} = x^k + \lambda^k [P_{C_1}(x^k - \nabla f(x^k)) - x^k] \quad \lambda^k > 0 \text{ pour tout } k > 0 \quad (1)$$

Compte-tenu du fait que $\nabla f(x^k) = (x^k - P_{C_2}(x^k))$ la relation (1) prend la forme suivante :

$$x^{k+1} = x^k + \lambda^k [P_{C_1}[P_{C_2}(x^k)] - x^k] \quad \lambda^k > 0 \quad \forall k > 0 \quad (2)$$

Ainsi on obtient :

Algorithme 1:

1) A l'itération $k=0$ on est en $x^0 \in C_1$.

2) A l'itération courante k on est en x^k . L'itéré x^{k+1} est donné par :

$$x^{k+1} = x^k + \lambda^k [P_{C_1}[P_{C_2}(x^k)] - x^k] \quad \lambda^k > 0 \quad \forall k > 0$$

V.2.4. CAS PARTICULIER DE L'ALGORITHME DU GRADIENT PROJETÉ :

L'algorithme du gradient projeté décrit précédemment peut être particularisé pour donner la méthode de projections successives qui fournit une stratégie optimale de choix du pas de déplacement λ^k dans le cas où l'un des deux convexes est une variété linéaire.

V.2.4.1. ALGORITHME DES PROJECTIONS SUCCESSIVES :

Appliquons au problème (P_1) la méthode décrite ci-dessus en prenant à chaque étape $\lambda^k = 1$ il vient :

Pour un point $x^0 \in C_1$ arbitraire on a :

$$x^{k+1} = P_{C_1}[P_{C_2}(x^k)]$$

Ce qui est équivalent à :

Algorithme 2 :

Partons d'un point $x^0 \in C_1$, le passage de (y^k, x^k) à (y^{k+1}, x^{k+1}) s'effectue de la façon suivante :

1) on définit y^{k+1} par

$$y^{k+1} = P_{C_2}(x^k) \tag{3}$$

2) puis on calcule x^{k+1} par

$$x^{k+1} = P_{C_1}(y^{k+1}) \tag{4}$$

Cet algorithme consiste donc à projeter alternativement sur chacun des convexes C_1 et C_2 . On reconnaît donc dans cet algorithme, la méthode de projections successives, voir [21].

THEOREME 1 [21] : Si C_1 , ou C_2 , est borné, l'ensemble S des optima, défini par $S = \{(x^*, y^*) \in C = C_1 \cap C_2 ; d(x^*, y^*) = d(C_1, C_2)\}$ est convexe, fermé, et non vide.

THEOREME 2 : Si $C_1 \cap C_2 \neq \emptyset$, alors les deux suites (x^k) et (y^k) définies par l'algorithme 2 convergent vers $x^* = y^* \in C = C_1 \cap C_2$.

Si $C_1 \cap C_2 = \emptyset$ et C_1 (ou C_2) est borné, alors la suite (x^k, y^k) définie par l'algorithme 2 converge vers $(x^*, y^*) \in C_1 \times C_2$ et :

$$\lim_{k \rightarrow \infty} \|x^k - y^k\| = d(C_1, C_2)$$

Preuve :

Avant de présenter la démonstration du théorème, on donne quelques propriétés de la projection d'un point sur un convexe fermé C , voir A.AUSLENDER[3].

Pour tout $x, y \in \mathbb{R}^n$ on a:

$$\|P_C(x) - P_C(y)\| \leq \|x - y\| \tag{5}$$

D'autre part tout $x \in C$ vérifie :

$$\|y - x\|^2 \geq \|y - P_C(y)\|^2 + \|x - P_C(y)\|^2 \tag{6}$$

Les equations (3) et (4) de l'algorithme 2 donnent :

$$x^{k+1} = P_{C_1}(y^k) \text{ et } y^{k+1} = P_{C_2}(x^{k+1})$$

Lemme :

$$\|x^{k+1} - y^k\|^2 \geq \|x^{k+1} - y^{k+1}\|^2 \geq \|x^{k+2} - y^{k+1}\|^2$$

Preuve : En effet;

$$\|x^{k+1} - y^{k+1}\|^2 = \|P_{C_2}(x^{k+1}) - y^k\|^2 \leq \|x^{k+1} - y\|^2 \text{ pour tout } y \in C_2.$$

En particulier pour $y^k \in C_2$ on vérifie les deux inégalités suivantes :

$$\|x^{k+1} - y^{k+1}\|^2 \leq \|x^{k+1} - y^k\|^2 \text{ pour tout } k \geq 0 \quad (7)$$

$$\|x^{k+2} - y^{k+1}\|^2 \leq \|x^{k+1} - y^{k+1}\|^2 \text{ pour tout } k \geq 0 \quad (8)$$

On conclut de (7) et (8) que les deux suites ($\|x^k - y^k\|^2$) et ($\|x^{k+1} - y^k\|^2$) sont décroissantes minorées par zéro, elles convergent donc et ont la même limite. De plus la propriété (6) permet de vérifier que :

$$\|y^k - x^k\|^2 \geq \|y^k - x^{k+1}\|^2 + \|x^k - x^{k+1}\|^2 \quad (9)$$

en passant à la limite et en tenant compte de (3) et (4) on obtient :

$$\lim_{k \rightarrow \infty} \|x^k - x^{k+1}\| = 0 \quad (10)$$

Soit x^* une valeur d'adhérence de la suite (x^k) . Elle existe toujours car la suite (x^k) est bornée, et il existe donc une sous-suite qui converge vers x^* . On a donc :

$$\forall \varepsilon > 0, \exists p_0 \text{ tel que } \forall p > p_0 \text{ on a } \|x^{k_p} - x^*\| < \varepsilon$$

Compte-tenu de (5) on aura:

$$\forall \varepsilon > 0 \exists p_0 \text{ tel que } \forall p \geq p_0 \text{ on a } \|P_{C_2}(x^{k,p}) - P_{C_2}(x^*)\| \leq \|x^{k,p} - x^*\| < \varepsilon, \text{ or}$$

$$\|P_{C_2}(x^{k,p}) - P_{C_2}(x^*)\| = \|y^{(k,p)+1} - P_{C_2}(x^*)\|, \text{ d'où}$$

$$\forall \varepsilon > 0 \exists p_0 \text{ tel que } \forall p \geq p_0 \text{ on a } \|y^{(k,p)+1} - P_{C_2}(x^*)\| < \varepsilon,$$

et donc $P_{C_2}(x^*) = y^*$ est une valeur d'adhérence de la suite (y^k) (11)

On a donc :

$$\forall \varepsilon > 0, \exists p_0 \text{ tel que } \forall p \geq p_0 \text{ on a } \|x^{k,p} - x^*\| < \varepsilon$$

Compte-tenu de (5) on aura.

$$\forall \varepsilon > 0 \exists p_0 \text{ tel que } \forall p \geq p_0 \text{ on a } \|P_{C_2}(x^{k,p}) - P_{C_2}(x^*)\| \leq \|x^{k,p} - x^*\| < \varepsilon, \text{ or}$$

$$\|P_{C_2}(x^{k,p}) - P_{C_2}(x^*)\| = \|y^{(k,p)+1} - P_{C_2}(x^*)\|, \text{ d'où}$$

$$\forall \varepsilon > 0 \exists p_0 \text{ tel que } \forall p \geq p_0 \text{ on a } \|y^{(k,p)+1} - P_{C_2}(x^*)\| < \varepsilon,$$

et donc $P_{C_2}(x^*) = y^*$ est une valeur d'adhérence de la suite (y^k) (11)

Comme $P_{C_1}P_{C_2}$ est un application faiblement contractante qui admet donc un point fixe, on vérifie, puisque $x^{k+1} = P_{C_1}(y^k) = P_{C_1}[P_{C_2}(x^k)]$:

$$x^* = P_{C_1}[P_{C_2}(x^*)] = P_{C_1}(y^*) \quad (12)$$

D'après le théorème de caractérisation, les relations (11) et (12) traduisent le fait que (x^*, y^*) est une solution du problème.

Lemme : $0 \leq \|x^{k+1} - x^*\|^2 \leq \|x^k - x^*\|^2$ et $\|y^{k+1} - y^*\|^2 \leq \|y^k - y^*\|^2$.

Preuve : En effet, on a

$$\|x^{k+1} - x^*\|^2 = \|P_{C_1}(y^k) - P_{C_1}(y^*)\|^2 \leq \|y^k - y^*\|^2 = \|P_{C_2}(x^k) - P_{C_2}(x^*)\|^2 \leq \|x^k - x^*\|^2 \quad (13)$$

$$\|y^{k+1} - y^*\|^2 \leq \|y^k - y^*\|^2 \quad (14)$$

On constate que la suite $(\|x^k - x^*\|^2)$ ainsi que $(\|y^k - y^*\|^2)$ sont décroissantes, et d'après (11)

$$\forall \varepsilon > 0 \exists p_0 \text{ tel que } \forall p \geq p_0 \text{ on a } \|x^{k,p} - x^*\| < \varepsilon$$

La relation (14) permet alors d'écrire :

$$\|x^k - x^*\|^2 \leq \|x^{k,p} - x^*\|^2 \quad \forall k \geq k, p, \text{ d'où}$$

$$\forall \varepsilon > 0 \exists k, p \text{ tel que } \forall k \geq k, p \text{ on a } \|x^k - x^*\| < \varepsilon \quad (15)$$

L'expression (15) traduit le fait que la suite (x^k) converge toute entière vers

Soit le problème :

$$\min\{1/2\|x - P_{C_2}(x)\|^2 ; x \in C_1\} \quad (P_4)$$

Le processus itératif décrit au paragraphe (1.2), pour résoudre le problème (P₄), se définit ainsi

$$x^{k+1} = x^k + \lambda^k [P_{C_1}(P_{C_2}(x^k)) - x^k], \text{ avec}$$

$$\lambda^k = \operatorname{argmin}\{f(x^k + \lambda d^k) ; 0 \leq \lambda \leq \lambda_{\max}\} \quad \text{où } d^k = P_{C_1}(P_{C_2}(x^k)) - x^k$$

$$\lambda_{\max} = \max\{\lambda ; x^k + \lambda [P_{C_1}(P_{C_2}(x^k)) - x^k] \in C_1\}$$

$$\lambda_{\max} = \max\{\lambda ; (1-\lambda)x^k + \lambda P_{C_1}(P_{C_2}(x^k)) \in C_1\},$$

il est alors facile de vérifier que $\lambda_{\max} = 1$ dans ce cas là. Posons

$$x^k = P_{C_1}(P_{C_2}(x^k)).$$

Pour déterminer λ^k on minimise la fonction suivante

$$f(x^k + \lambda d^k) = 1/2 \|x^k + \lambda(x^k - x^k) - P_{C_2}(x^k) - \lambda[P_{C_2}(x^k) - P_{C_2}(x^k)]\|^2$$

$$f(x^k + \lambda d^k) = 1/2 \|(1-\lambda)[x^k - P_{C_2}(x^k)] + \lambda[x^k - P_{C_2}(x^k)]\|^2,$$

posons $u^k = x^k - P_{C_2}(x^k)$ et $v^k = x^k - P_{C_2}(x^k)$,

on aura donc

$$f(x^k + \lambda d^k) = 1/2(1-\lambda)^2 \|u^k\|^2 + (1-\lambda)\lambda \langle u^k, v^k \rangle + 1/2\lambda^2 \|v^k\|^2$$

La fonction $\lambda \rightarrow f(x^k + \lambda d^k)$ atteint son minimum en

$$\lambda^k = \frac{\|u^k\|^2 - \langle u^k, v^k \rangle}{\|u^k\|^2 - 2\langle u^k, v^k \rangle + \|v^k\|^2} \quad (**)$$

Où bien

$$\lambda^k = \frac{\|x^k - P_{C_2}(x^k)\|^2 - \langle x^k - P_{C_2}(x^k), x^k - P_{C_2}(x^k) \rangle}{\|x^k - P_{C_2}(x^k) - \tilde{x}^k + P_{C_2}(x^k)\|^2}$$

Le minimum λ^k ainsi déterminé de La fonction $\lambda \rightarrow f(x^k + \lambda d^k)$ doit être compris entre 0 et 1. Or, on montre que λ^k donnée par la relation (**) est supérieur à 1 pour tout $k=0, 1, \dots$, i.e., $\lambda^k \geq 1$ pour tout $k \geq 0$.

En effet;

$$\lambda^k \geq 1 \Leftrightarrow \|u^k\|^2 - \langle u^k, v^k \rangle \geq \|u^k\|^2 - 2\langle u^k, v^k \rangle + \|v^k\|^2 \Leftrightarrow \|v^k\|^2 - \langle u^k, v^k \rangle \leq 0$$

$$\Leftrightarrow \|v^k\|^2 \leq \|u^k\| \cdot \|v^k\| \Leftrightarrow \|v^k\| \leq \|u^k\| \Leftrightarrow \|x^k - P_{C_2}(x^k)\| \leq \|x^k - P_{C_2}(x^k)\|$$

Cette dernière inégalité est vraie pour tout $k \geq 0$. On a donc :

$$\lambda^k \geq 1 \text{ pour tout } k \geq 0.$$

Alors la fonction $\lambda \rightarrow f(x^k + \lambda d^k)$ atteint son minimum sur $[0, 1]$ pour la valeur $\lambda = 1$. On a donc $\lambda^k = 1 \forall k > 0$. D'où le processus itératif :

$$x^{k+1} = x^k + d^k \text{ avec } d^k = x^k - x^k$$

Ainsi on aura:

$$x^{k+1} = x^k = P_{C_1}(P_{C_2}(x^k))$$

Conclusion : On conclut que la méthode des projections successives pour la recherche de la distance de deux convexes fermés dans le cas où l'un d'eux est une variété linéaire est la méthode pour laquelle le choix du paramètre de relaxation est optimal.

V.3. APPLICATION DE LA METHODE DE DECOMPOSITION A LA RECHERCHE DE L'INTERSECTION DE (OU DE LA DISTANCE ENTRE) DEUX CONVEXES FERMES :

La méthode connue pour ce type de problème est celle des projections successives décrite au paragraphe V.2.4.1. Dans cette partie, on développe une méthode de projections du type barycentrique en formulant le problème dans un espace produit et en lui appliquant la méthode de décomposition[2].

V.3.1. POSITION DU PROBLEME :

Soit le problème :

Trouver $(x_1^*, x_2^*) \in C_1 \times C_2$ tel que

$$1/2 \|x_1^* - x_2^*\|^2 = \min\{1/2 \|x_1 - x_2\|^2 ; (x_1, x_2) \in C_1 \times C_2\} \quad (P_{2,1})$$

$C_i \subset \mathbb{R}^n \quad i=1,2$ deux convexes fermés.

Caractérisation de l'ensemble de solution du problème $(P_{2,1})$:

On a l'équivalence entre les 3 propositions suivantes :

- 1) $(x_1^*, x_2^*) \in C_1 \times C_2 \quad : d(x_1^*, x_2^*) = d(C_1, C_2)$.
- 2) $(x_1^*, x_2^*) \in C_1 \times C_2 \quad : d(x_1^*, x_2^*) = d(C_1, x_2^*) = d(x_1^*, C_2)$.
- 3) $x_1^* = P_{C_1}(x_2^*)$ et $x_2^* = P_{C_2}(x_1^*)$

Posons $\mathcal{C} = C_1 \times C_2$ et $\mathcal{D} = \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^n ; x = y\}$.

Alors, dans l'espace produit, le problème $(P_{2,1})$ est équivalent, au problème suivant :

Trouver $(x^*, y^*) \in \mathcal{C} \times \mathcal{D}$ tel que

$$1/2 \|x^* - y^*\|^2 = \min\{1/2 \|x - y\|^2 ; (x, y) \in \mathcal{C} \times \mathcal{D}\}$$

On désigne les éléments de l'espace produit $\mathbb{R}^n \times \mathbb{R}^n$ par $\mathcal{X} = (x_1, x_2)$.

Le problème $(P_{2,1})$ s'écrit sous la forme :

Trouver $(x_1^*, x_2^*, y^*) \in C = C_1 \times C_2 \times \mathbb{R}^n$ tel que :

$$\min_{1 \leq i \leq 2} \lambda_i \|x_i^* - y^*\|^2 = \min_{1 \leq i \leq 2} \{1/2 \sum \lambda_i \|x_i - y\|^2 ; (x_1, x_2, y) \in C\}$$

Posons

$$f(x_1, x_2, y) = \frac{1}{2} \sum_{i=1}^2 \lambda_i \|x_i - y\|^2$$

$f(\cdot)$ est une fonction à valeurs réelles définie sur $\mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n$.

La recherche de l'intersection de deux convexes ou -si l'intersection est vide- de leur distance, revient à résoudre le problème d'optimisation suivant dans l'espace produit $\mathcal{C} = \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n$:

$$\text{Trouver } u^* \in C \text{ tel que } f(u^*) = \min\{f(u) ; u \in C\} \quad (P_{3,2})$$

Où $u^* = (x_1^*, x_2^*, y^*)$ et $u = (x_1, x_2, y)$.

La méthode de décomposition pour résoudre $(P_{2,1})$ consiste à construire à partir d'un élément arbitraire $u^0 \in \mathcal{C}$ une suite u^0, u^1, \dots, u^k d'éléments de \mathcal{C} qui convergent vers une solution u^* du problème $(P_{3,2})$.

On note:

$$u^k = (x_1^k, x_2^k, y^k) \text{ où } x_i^k \in C_i \text{ pour } i=1,2 \text{ et } y^k \in \mathbb{R}^n,$$

$$f_{1,k}(x_1) = f(x_1, x_2^k, y^k) \quad \text{et} \quad f_{1,k}(x_1) = f(x_1^{k+1}, x_2, y^k),$$

$$\text{et} \quad f_{3,k}(y) = f(x_1^{k+1}, x_2^{k+1}, y)$$

Pour passer de u^k à u^{k+1} , La méthode consiste à déterminer successivement les éléments x_1^{k+1} , x_2^{k+1} et y^{k+1} à partir de x_1^k, x_2^k, y^k en effectuant une seule itération d'une méthode destinée à résoudre le problème suivant :

$$\min\{f_{i,k}(x_i) ; x_i \in C_i\} \quad (P_{i,k}) \text{ pour } i=1,2,$$

et pour $i=3$ le problème :

$$\min\{f_{3,k}(y) ; y \in \mathbb{R}^n\} \quad (P_{3,k})$$

Les sous-problèmes résolus lors de chacun des pas de l'algorithme de décomposition, s'écrivent ainsi:

Trouver $x_1^{k+1} \in C_1$ tel que

$$f_{1,k}(x_1^{k+1}) = \min\{\frac{1}{2}\lambda_1\|x_1 - y^k\|^2 + \frac{1}{2}\lambda_2\|x_2^k - y^k\|^2 ; x_1 \in C_1\}$$

Trouver $x_2^{k+1} \in C_2$ tel que

$$f_{2,k}(x_2^{k+1}) = \min\{\frac{1}{2}\lambda_1\|x_1^{k+1} - y^k\|^2 + \frac{1}{2}\lambda_2\|x_2 - y^k\|^2 ; x_2 \in C_2\}$$

et trouver $y^{k+1} \in \mathbb{R}^n$ tel que

$$f_{3,k}(y) = \min\{\frac{1}{2}\sum_{i=1}^2 \lambda_i \|x_i^{k+1} - y\|^2 ; y \in \mathbb{R}^n\}$$

On vérifie alors pour le sous-problème $(P_{1,k})$ (resp $(P_{2,k})$) que le point $x_1^{k+1} \in C_1$ (resp $x_2^{k+1} \in C_2$) qui assure le minimum de $f_{1,k}$ (resp $f_{2,k}$) sur le convexe C_1 (resp C_2) n'est autre que la projection du point y^k sur C_1 (resp C_2), c'est à dire :

$$x_i^{k+1} = P_{C_i}(y^k) \text{ pour } i=1,2$$

une fois les x_i^{k+1} déterminés, le minimum du problème $(P_{3,k})$ est atteint au point y^{k+1} qui annule le gradient de la fonction $f_{3,k}$ c'est à dire :

$$\nabla f_{3,k}(y^{k+1}) = \sum_{i=1}^2 \lambda_i (y^{k+1} - x_i^{k+1}) = 0$$

On obtient alors :

$$y^{k+1} = \sum_{i=1}^2 \lambda_i x_i^{k+1} = \sum_{i=1}^2 \lambda_i P_{C_i}(y^k) \text{ avec } \sum_{i=1}^2 \lambda_i = 1 \text{ et } \lambda_i > 0.$$

Ainsi l'algorithme est conforme à la procédure suivante:

partant d'un $y^0 \in \mathbb{R}^n$ choisi arbitrairement, on construit de façon récurrente les suites (x_1^k) , (x_2^k) et (y^k) dans C_1 , C_2 et \mathbb{R}^n respectivement de la façon suivante :

supposons calculé y^k , alors :

1) à partir de y^k on définit x_1^{k+1} et x_2^{k+1} par la relation :

$$x_1^{k+1} = P_{C_1}(y^k)$$

$$x_2^{k+1} = P_{C_2}(y^k)$$

2) à partir x_1^{k+1} et x_2^{k+1} on définit y^{k+1} par la relation :

$$y^{k+1} = \sum_{i=1}^2 \lambda_i x_i^{k+1} = \sum_{i=1}^2 \lambda_i P_{C_i}(y^k) \text{ avec } \sum_{i=1}^2 \lambda_i = 1 \text{ et } \lambda_i > 0.$$

THEOREME 1 : Si $C_1 \cap C_2 \neq \emptyset$, les trois suites (x_1^k) , (x_2^k) et (y^k) définies par l'algorithme ci-dessus convergent vers $x_1^* = x_2^* = y^* \in C_1 \cap C_2$.

Si $C_1 \cap C_2 = \emptyset$ et C_1 (ou C_2) est borné, alors la suite (y^k) (resp (x_1^k, x_2^k)) converge vers $y^* \in \mathbb{R}^n$ (resp $(P_{C_1}(y^*), P_{C_2}(y^*)) \in C_1 \times C_2$) tel que :

$$\lim_{k \rightarrow \infty} \|x_1^k - x_2^k\|^2 = d(C_1, C_2) = d(P_{C_1}(y^*), P_{C_2}(y^*)).$$

Preuve : voir THEOREME 1 du paragraphe V.5.2.

V.4. METHODE DU GRADIENT PROJETE RELATIVE A UNE INTERSECTION DE CONVEXES :

Le propos de cette partie est de montrer, à travers la représentation de l'intersection dans l'espace produit, qu'on est amené à résoudre un problème d'optimisation convexe du même type que celui du paragraphe V.2. auquel on applique la méthode du gradient projeté.

V.4.1. NOUVELLE FORMULATION DU PROBLEME :

Soit le problème :

$$(P_{3,1}) \text{ Trouver } x \in \mathbb{R}^n \text{ tel que } x \in \bigcap_{i=1}^m C_i$$

Où les C_i sont des convexes fermés.

On va transformer le problème $(P_{3,1})$ dans l'espace produit et pour cela on donne les notations suivantes.

On munit \mathbb{R}^n du produit scalaire pondéré qu'on note par $(\cdot, \cdot)_i = \lambda_i (\cdot, \cdot)$ où (\cdot, \cdot) est le produit scalaire usuel sur \mathbb{R}^n avec

$$\sum_{i=1}^m \lambda_i = 1 \text{ et } \lambda_i > 0 \text{ pour } i=1, \dots, m.$$

Soient $C = C_1 \cap C_2 \cap \dots \cap C_m$ et $\mathfrak{H} = \mathbb{R}^n \times \mathbb{R}^n \times \dots \times \mathbb{R}^n$ (m fois).

L'espace \mathfrak{H} ainsi défini est doté de la topologie produit par le produit scalaire suivant

$$((\dots)) = \sum_{i=1}^m \lambda_i (\dots)$$

et de la norme qui lui est associée notée $||| \cdot |||$.

$\mathfrak{C} = C_1 \times C_2 \times \dots \times C_m$. On a l'équivalence suivante:

$$y \in C = \bigcap C_i \Leftrightarrow y \in C_1, y \in C_2, \dots, y \in C_m \Leftrightarrow (y, y, \dots, y) \in \mathfrak{C} = C_1 \times C_2 \times \dots \times C_m.$$

Ceci nous amène à introduire la définition :

DEFINITION 1 :

On appelle convexe diagonal \mathfrak{D} le sous-espace vectoriel de \mathfrak{H} défini par :

$$\mathfrak{D} = \{ (x, x, \dots, x) \in \mathfrak{H} ; x \in \mathbb{R}^n \}.$$

On note les éléments de \mathfrak{H} par $\mathfrak{X} = (x_1, x_2, \dots, x_m)$ avec $x_i \in \mathbb{R}^n$ pour $i=1, \dots, m$.

PROPOSITION 1 : Si $x \in C$ alors $\mathcal{X}=(x,x,\dots,x) \in \mathcal{C} \cap \mathcal{D}$.

Preuve : $x \in C \Leftrightarrow x \in C_1, x \in C_2, \dots, x \in C_m \Leftrightarrow (x,x,\dots,x) \in \mathcal{C} \Leftrightarrow (x,x,\dots,x) \in \mathcal{C} \cap \mathcal{D}$
par définition de \mathcal{D} .

PROPOSITION 2 : 1) La projection d'un point $\mathcal{X}=(x,x,\dots,x) \in \mathcal{D}$ sur le convexe produit \mathcal{C} a pour composantes les projections de x sur chacun des convexes.

$$P_{\mathcal{C}}(\mathcal{X})=(P_{C_1}(x), \dots, P_{C_m}(x)).$$

2) La projection d'un point $\mathcal{X}=(x_1, x_2, \dots, x_m)$ sur le convexe diagonal \mathcal{D} a pour composantes le barycentre pondéré des m points x_1, x_2, \dots, x_m .

$$P_{\mathcal{D}}(\mathcal{X})=(\sum_{i=1}^m \lambda_i x_i, \sum_{i=1}^m \lambda_i x_i, \dots, \sum_{i=1}^m \lambda_i x_i)$$

Preuve :

1) Soit $\mathcal{X} \in \mathcal{D}$ par définition de la projection sur un convexe on a

$$\|P_{\mathcal{C}}(\mathcal{X}) - \mathcal{X}\|^2 = \inf\{\|y - \mathcal{X}\|^2 ; y \in \mathcal{C}\} = \inf\{\sum_{i=1}^m \lambda_i \|y_i - x\|^2 ; y_i \in C_i \text{ pour } i=1, \dots, m\}$$

Or $\inf\{\lambda_i \|y_i - x\|^2 ; y_i \in C_i\} = \lambda_i \|P_{C_i}(x) - x\|^2$, d'où

$$\|P_{\mathcal{C}}(\mathcal{X}) - \mathcal{X}\|^2 = \sum_{i=1}^m \lambda_i \|P_{C_i}(x) - x\|^2 = \|y^* - \mathcal{X}\|^2 = \sum_{i=1}^m \lambda_i \|y_i^* - x\|^2$$

$$\text{D'où } y^* = (y_1^*, y_2^*, \dots, y_m^*) = P_{\mathcal{C}}(\mathcal{X}) = (P_{C_1}(x), P_{C_2}(x), \dots, P_{C_m}(x))$$

2) Soit $y \in \mathcal{C}$ de même que 1) on a

$$\|P_{\mathcal{D}}(y) - y\|^2 = \inf\{\|x - y\|^2 ; x \in \mathcal{D}\} = \inf\{\sum_{i=1}^m \lambda_i \|y_i - x\|^2 ; x \in \mathbb{R}^n\}$$

On a à minimiser une fonction convexe sur tout l'espace, une condition nécessaire et suffisante pour que x soit solution est que le gradient de $f(x) = \sum_{i=1}^m \lambda_i \|y_i - x\|^2$ soit nul c'est à dire :

$$2 \sum_{i=1}^m \lambda_i (y_i - x) = 0 \Rightarrow \sum_{i=1}^m \lambda_i y_i = \sum_{i=1}^m \lambda_i x = x \Rightarrow x = \sum_{i=1}^m \lambda_i y_i$$

d'où

$$P_{\mathcal{D}}(y) = x^* = (\sum_{i=1}^m \lambda_i y_i, \sum_{i=1}^m \lambda_i y_i, \dots, \sum_{i=1}^m \lambda_i y_i)$$

Considérons le problème

$$(P_{3,2}) : \text{Trouver } (x^*, y^*) \in \mathcal{E} \times \mathcal{D} \text{ tel que}$$

$$\|x^* - y^*\|^2 = \min\{\|x - y\|^2; (x, y) \in \mathcal{E} \times \mathcal{D}\}$$

Caractérisation de la solution du problème $(P_{3,2})$:

On a l'équivalence entre ces 3 propositions

- 1) $(x^*, y^*) \in \mathcal{E} \times \mathcal{D} : d(x^*, y^*) = d(\mathcal{E}, \mathcal{D})$
- 2) $(x^*, y^*) \in \mathcal{E} \times \mathcal{D} : d(x^*, y^*) = d(x^*, \mathcal{D}) = d(y^*, \mathcal{E})$
- 3) $x^* = P_{\mathcal{E}}(y^*) \text{ et } y^* = P_{\mathcal{D}}(x^*)$

Ou encore en exprimant cette caractérisation dans l'espace \mathbb{R}^n :

$$-x_i^* = P_{C_i}(y^*)$$

$$-y^* = \sum_{1 \leq i \leq m} \lambda_i x_i^*$$

avec $x^* = (x_1^*, x_2^*, \dots, x_m^*)$ et $y^* = (y^*, y^*, \dots, y^*)$

PROPOSITION 3 : Sous l'hypothèse $(H^0) : C = \bigcap_{1 \leq i \leq m} C_i \neq \emptyset$. Le problème $(P_{3,2})$ est équivalent au problème :

$$(P_{3,1}) : \text{Trouver } x \in \mathbb{R}^n \text{ tel que } x \in \bigcap_{1 \leq i \leq m} C_i$$

Preuve :

Si $C \neq \emptyset$ alors il existe au moins un $x \in C = \bigcap C_i$ c'est à dire $x \in C_1, x \in C_2, \dots, x \in C_m$ ce qui implique $(x, x, \dots, x) \in \mathcal{E} \cap \mathcal{D}$ d'où $d(\mathcal{E}, \mathcal{D}) = 0$ ce qui veut dire que le minimum de $1/2 \|x - y\|^2$ sur $\mathcal{E} \times \mathcal{D}$ sera nul d'où le fait que $[(x, x, \dots, x), (x, x, \dots, x)] \in \mathcal{E} \times \mathcal{D}$ est solution du problème $(P_{3,2})$ avec $x \in C = \bigcap C_i$.

Donc, trouver $x \in C = \bigcap C_i$ est équivalent à :

$$\min\{1/2 \|x - y\|^2; x \in \mathcal{E}, y \in \mathcal{D}\},$$

où $x = (x_1, x_2, \dots, x_m)$ et $y = (y, y, \dots, y)$.

On a donc une équivalence entre le problème de la recherche de l'intersection de convexes et le problème $(P_{3,2})$.

Or le problème $(P_{3,2})$ est un problème de la recherche de l'intersection de \mathcal{E} et \mathcal{D} dans l'espace produit. De plus il faut remarquer que l'ensemble \mathcal{D} est sous-espace vectoriel dont le projecteur qui lui est associé est un opérateur linéaire.

Le problème $(P_{3,2})$ se présente sous deux formes équivalentes :

$$(P_{3,3}) : \min\{ 1/2\|x - P_{\mathcal{E}}(x)\|^2 ; x \in \mathcal{D}\}$$

$$(P_{3,4}) : \min\{ 1/2\|y - P_{\mathcal{D}}(y)\|^2 ; y \in \mathcal{E}\}$$

En appliquant la méthode du gradient projeté décrite au paragraphe (1) aux deux problèmes on aura :

pour le problème $(P_{3,3})$ le processus itératif de la méthode du gradient projeté est comme suit :

étant donné x^0 un élément arbitraire de \mathbb{R}^n , l'algorithme génère une suite (x^k) dont l'itéré x^{k+1} est donné par le relation suivante :

$$x^{k+1} = x^k + t^k [P_{\mathcal{D}} P_{\mathcal{E}}(x^k) - x^k] \quad (1)$$

Ou encore en exprimant la relation (1) dans l'espace \mathbb{R}^n on aura

$$x^{k+1} = x^k + t^k \sum_{1 \leq i \leq m} \lambda_i [P_{C_i}(x^k) - x^k] \quad \text{avec} \quad \sum_{1 \leq i \leq m} \lambda_i = 1 \quad \text{et} \quad \lambda_i > 0 \quad \mu^k > 0.$$

Donc c'est le processus itératif pour la détermination d'un point appartenant à l'intersection des convexes fermés donnés.

Le processus itératif de la méthode du gradient projeté, pour la résolution du problème $(P_{3,4})$, est comme suit :

$$y^{k+1} = y^k + t^k [P_{\mathcal{E}}(P_{\mathcal{D}}(y^k)) - y^k] \quad (1.1.2)$$

En exprimant la relation (1.1.2) dans l'espace \mathbb{R}^n on obtient :

$$y_i^{k+1} = y_i^k + t^k [P_{C_i}(\sum_{1 \leq j \leq m} \lambda_j y_j^k) - y_i^k] \quad \text{pour } i=1, \dots, m.$$

C'est dans ce cas qu'on peut avoir un choix optimal du paramètre de relaxation μ^k . D'après l'étude faite au paragraphe(1.3.2) la stratégie optimale pour le choix du pas est de prendre $\mu^k=1$ pour tout $k>0$. On aura donc :

$$y_i^{k+1}=P_{C_i}(\sum \lambda_i y_i^k) \quad \text{pour } i=1,\dots,m.$$

Si on pose $x^k=\sum \lambda_i y_i^k$ on aura $y_i^{k+1}=P_{C_i}(x^k)$ et $x^{k+1}=\sum \lambda_i y_i^{k+1}=\sum \lambda_i P_{C_i}(x^k)$.

On remarque que ce n'est autre que la méthode du barycentre.

V.5. APPLICATION DE LA METHODES DE DECOMPOSITION A LA RECHERCHE DE L'INTERSECTION D'UNE FAMILLE DE CONVEXES FERMES :

A travers la représentation de l'intersection des convexes dans l'espace produit, on s'aperçoit qu'on est en présence d'un problème de minimisation d'une fonction différentiable dans l'espace produit qu'on se propose de le résoudre par la méthode de décomposition[2].

V.5.1. POSITION DU PROBLEME :

Soit le problème :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } x \in C_1 \cap C_2 \cap \dots \cap C_m, \quad (P_{5,1})$$

pour $i=1, \dots, m$ les $C_i \subset \mathbb{R}^n$ sont des convexes fermés.

Dans le paragraphe précédent on a montré que $(P_{5,1})$ est équivalent dans l'espace produit au problème suivant :

Trouver $(x^*, y^*) \in \mathcal{E} \times \mathcal{D}$ tel que

$$1/2 \|x^* - y^*\|^2 = \min \{ 1/2 \|x - y\|^2 ; (x, y) \in \mathcal{E} \times \mathcal{D} \}$$

Qui peut s'écrire sous la forme :

Trouver $(x_1^*, x_2^*, \dots, x_m^*, y^*) \in C = C_1 \times C_2 \times \dots \times C_m \times \mathbb{R}^n$ tels que :

$$\frac{1}{2} \sum_{i=1}^m \lambda_i \|x_i^* - y^*\|^2 = \min \left\{ \frac{1}{2} \sum_{i=1}^m \lambda_i \|x_i - y\|^2 ; (x_1, x_2, \dots, x_m, y) \in C \right\}.$$

Posons $f(x_1, x_2, \dots, x_m, y) = 1/2 \sum \lambda_i \|x_i - y\|^2$. $f(\cdot)$ est une fonction à valeurs réelles définie sur $\mathbb{R}^n \times \mathbb{R}^n \times \dots \times \mathbb{R}^n$ ($m+1$ fois).

La recherche d'un point commun à une famille de convexes fermés revient donc à résoudre le problème d'optimisation, dans l'espace produit \mathcal{X} , qui s'écrit ainsi:

$$\text{Trouver } u^* \in C \text{ tel que } f(u^*) = \min \{ f(u) ; u \in C \}, \quad (P_{5,2})$$

où $u^* = (x_1^*, x_2^*, \dots, x_m^*, y^*)$ et $u = (x_1, \dots, x_m, y)$.

La méthode de décomposition qu'on propose pour résoudre $(P_{5,2})$ consiste à construire à partir d'un élément arbitraire $u^0 \in \mathcal{X}$ une suite u^0, u^1, \dots, u^k d'éléments de \mathcal{X} qui convergent vers une solution u^* du problème $(P_{5,2})$.

On note:

$$u^k = (x_1^k, x_2^k, \dots, x_m^k, y^k) \text{ où } x_i^k \in C_i \text{ pour } i=1, \dots, m \text{ et } y^k \in \mathbb{R}^n;$$

et $f_{i,k}(x_i) = f(x_1^{k+1}, x_2^{k+1}, \dots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \dots, x_m^k, y^k)$ pour $i=1, \dots, m$

et $f_{m+1,k}(y) = f(x_1^{k+1}, x_2^{k+1}, \dots, x_m^{k+1}, y)$

Pour passer de u^k à u^{k+1} , La méthode consiste à déterminer successivement pour $i=1, \dots, m$ l'élément x_i^{k+1} et pour $i=m+1$ l'élément y^{k+1} à partir de $x_1^k, x_2^k, \dots, x_m^k, y^k$ en effectuant une seule itération d'une méthode de descente destinée à résoudre le problème suivant :

$$\min\{f_{i,k}(x_i) ; x_i \in C_i\}, \quad (P_{i,k}) \text{ pour } i=1, \dots, m$$

et pour $i=m+1$ le problème :

$$\min\{f_{m+1,k}(y) ; y \in \mathbb{R}^n\}. \quad (P_{m+1,k})$$

Les sous-problèmes résolus lors de chacun des pas de l'algorithme de décomposition, s'écrivent ainsi:

Trouver $x_i^{k+1} \in C_i$ tel que

$$f_{i,k}(x_i^{k+1}) = \min\left\{1/2 \sum_{j=1}^{i-1} \lambda_j \|x_j^{k+1} - y^k\|^2 + 1/2 \|x_i - y^k\|^2 + 1/2 \sum_{j=i+1}^m \lambda_j \|x_j^k - y^k\|^2 ; x_i \in C_i\right\}$$

et trouver $y^{k+1} \in \mathbb{R}^n$ tel que

$$f_{m+1,k}(y^{k+1}) = \min\left\{1/2 \sum_{i=1}^m \lambda_i \|x_i^{k+1} - y\|^2 ; y \in \mathbb{R}^n\right\} \quad (P_{m+1,k})$$

On vérifie alors pour les sous-problèmes $(P_{i,k})$ que le point $x_i^{k+1} \in C_i$ qui assure le minimum de $f_{i,k}$ sur le convexe C_i n'est autre que la projection du point y^k sur C_i , c'est à dire :

$$x_i^{k+1} = P_{C_i}(y^k) \text{ pour tout } i=1, \dots, m,$$

une fois les x_i^{k+1} déterminés, on résoud le problème $(P_{m+1,k})$ en définissant à partir de y^k la direction de descente $p_{k,m+1} = \nabla f_{m+1,k}(y^k)$ alors y^{k+1} sera :

$$y^{k+1} = y^k - \mu^k \nabla f_{m+1,k}(y^k)$$

Comme $\nabla f_{m+1,k}(y^k) = -\sum \lambda_i [x_i^{k+1} - y^k]$ et $x_i^{k+1} = P_{C_i}(y^k)$ pour tout $i=1, \dots, m$ on a alors :

$$y^{k+1} = y^k + \mu^k \sum_{1 \leq i \leq m} \lambda_i [P_{C_i}(y^k) - y^k] \quad \mu^k \geq 0,$$

ainsi l'algorithme est conforme à la procédure suivante :

partant d'un $y^0 \in \mathbb{R}^n$ choisi arbitrairement. Le passage de y^k à y^{k+1} s'effectue de la façon suivante:

Pour tout $i=1, \dots, m$ on calcule la projection de y^k sur le convexe C_i , i.e.,

$$x_i^{k+1} = P_{C_i}(y^k),$$

puis on détermine :

$$y^{k+1} = y^k + \mu^k \sum_{i=1}^m \lambda_i [x_i^{k+1} - y^k] = y^k + \mu^k \sum_{i=1}^m \lambda_i [P_{C_i}(y^k) - y^k] \quad \mu^k \geq 0.$$

PROPOSITION 1 : La résolution du problème $(P_{5.1})$ par la méthode de décomposition décrite ci-dessus est équivalente à résoudre le problème d'optimisation suivant :

$$\min \left\{ \sum_{i=1}^m \lambda_i \|P_{C_i}(y) - y\|^2 ; y \in \mathbb{R}^n \right\} \text{ avec } \sum_{i=1}^m \lambda_i = 1 \text{ et } \lambda_i > 0.$$

Par la méthode classique du gradient .

Remarque : L'équivalence est dans le sens que les deux méthodes donnent le même processus itératif.

Preuve : Le problème qu'on se propose de résoudre est :

$$\min \left\{ \sum_{i=1}^m \lambda_i \| P_{C_i}(y) - y \|^2 ; y \in \mathbb{R}^n \right\} \quad \text{où } \sum_{i=1}^m \lambda_i = 1 \quad \text{et } \lambda_i > 0 \quad (P_{5,2}).$$

Comme ce problème est sans contrainte et que sa fonction objectif est différentiable, on propose alors de le résoudre par la méthode classique du gradient qu'on décrit ci-dessous.

V.5.2. PRESENTATION DE LA METHODE DU GRADIENT :

Il s'agit d'une méthode destinée à résoudre un problème d'optimisation convexe sans contraintes où la fonction objectif est différentiable et qui procède de la façon suivante :

On part d'un point x^0 et on calcule le gradient $\nabla f(x^0)$ en x^0 . Comme $\nabla f(x^0)$ indique la direction de plus grande augmentation de f , on se déplace d'une quantité μ^0 dans la direction opposée au gradient, et on définit le point :

$$x^1 = x^0 - \mu^0 \nabla f(x^0)$$

La procédure est répétée et engendre x^0, x^1, \dots, x^k , suivant la relation :

$$x^{k+1} = x^k - \mu^k \nabla f(x^k)$$

Dans cette famille de méthodes, il convient de signaler les méthodes de gradient à pas prédéterminé dans lesquelles on choisit à priori les valeurs des déplacements μ^k .

Le problème $(P_{5,2})$ entre dans la classe des problèmes qui sont résolubles par cette famille de méthodes. Pour appliquer la dite méthode on a à

calculer le gradient de $f(y) = \sum_{i=1}^m \lambda_i \| P_{C_i}(y) - y \|^2$ qui est donné par :

$$\nabla f(y) = \sum_{i=1}^m \lambda_i (y - P_{C_i}(y)) \quad (*).$$

D'où le processus itératif pour la résolution du problème $(P_{5,2})$.

Algorithme 3 :

Choisir un point arbitraire $y^0 \in \mathbb{R}^n$, à l'itération k on est en y^k , l'itéré y^{k+1} est donné par :

$$y^{k+1} = y^k - \mu^k \nabla f(y^k)$$

Compte-tenu de la relation (*) on obtient :

$$y^{k+1} = y^k + \mu^k \sum_{1 \leq i \leq m} \lambda_i (P_{C_i}(y^k) - y^k) \quad \text{avec } \mu^k > 0 \text{ et } \sum_{1 \leq i \leq m} \lambda_i = 1 \text{ avec } \lambda_i > 0.$$

On voit qu'on a obtenu le même processus itératif que l'algorithme précédent avec $0 \leq \mu^k \leq 2 \quad \forall k \geq 0$.

THEOREME 1 : Si $\cap C_i \neq \emptyset$ et $0 \leq \mu^k \leq 2 \quad \forall k \geq 0$, la suite (y^k) définie par l'algorithme précédent converge vers un élément $y^* \in \cap C_i$ et on a :

$$\sum_{i=1}^m \lambda_i \|P_{C_i}(y^k) - y^k\|^2 \rightarrow 0 \text{ quand } k \rightarrow +\infty.$$

Preuve :

On a :

$$y^{k+1} = y^k + \mu^k \sum_{1 \leq i \leq m} \lambda_i (P_{C_i}(y^k) - y^k) \quad \text{avec } 0 \leq \mu^k \leq 2$$

$$\text{et } f(y) = \sum_{i=1}^m \lambda_i \|P_{C_i}(y^k) - y^k\|^2.$$

1^{er} point : Montrons que $f(y^{k+1}) \leq f(y^k) - (2/\mu^k - 1) \|y^{k+1} - y^k\|^2$
En effet,

$$\begin{aligned} \|P_{C_i}(y^{k+1}) - y^{k+1}\|^2 &\leq \|P_{C_i}(y^k) - y^k\|^2 \\ &= \|P_{C_i}(y^k) - y^k\|^2 + \|y^{k+1} - y^k\|^2 - \langle P_{C_i}(y^k) - y^k, y^{k+1} - y^k \rangle, \end{aligned}$$

sommons sur i ,

$$f(y^{k+1}) = \sum_{i=1}^m \lambda_i \|P_{C_i}(y^{k+1}) - y^{k+1}\|^2 \leq \sum_{i=1}^m \lambda_i \|P_{C_i}(y^k) - y^k\|^2 + \|y^{k+1} - y^k\|^2 - \sum_{i=1}^m \lambda_i \langle P_{C_i}(y^k) - y^k, y^{k+1} - y^k \rangle.$$

Or d'après (1) on a :

$$\sum_{i=1}^m \lambda_i [P_{C_i}(y^k) - y^k] = \frac{y^{k+1} - y^k}{\mu^k}.$$

On obtient donc :

$$f(y^{k+1}) \leq f(y^k) + \|y^{k+1} - y^k\|^2 - 2/\mu^k \|y^{k+1} - y^k\|^2 = f(y^k) - [(2/\mu^k) - 1] \|y^{k+1} - y^k\|^2 \quad (2)$$

2^{ème} point : Soit $F = \{x \in \mathbb{R}^n ; x = \sum \lambda_i P_{C_i}(x)\}$; montrons que $F = C = \cap C_i$.

1) $F \subset C$ évident.

2) $C \subset F$. Soit $z \in F$, $x \in C$ donc $P_{C_i}(x) = x$ et $\|P_{C_i}(x^k) - P_{C_j}(x^k)\|^2 = 0 \quad \forall i, j$.

On a :

$$0 = \langle x - \sum_{i=1}^m \lambda_i P_{C_i}(z), z - \sum_{i=1}^m \lambda_i P_{C_i}(z) \rangle = \langle \sum_{i=1}^m \lambda_i P_{C_i}(x) - \sum_{i=1}^m \lambda_i P_{C_i}(z), z - \sum_{i=1}^m \lambda_i P_{C_i}(z) \rangle$$

$$\leq - \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j \| P_{C_i}(z) - P_{C_j}(z) \|^2$$

D'où

$$\sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j \| P_{C_i}(z) - P_{C_j}(z) \|^2 \leq 0 \Leftrightarrow P_{C_i}(z) = P_{C_j}(z) \quad \forall i, j$$

Alors $z = \sum_{i=1}^m \lambda_i P_{C_i}(z) = P_{C_j}(z) \quad \forall j$ c'est à dire $z \in C_j \quad \forall j$; ce qui implique que

$z \in C$, d'où $F = C$.

3^{ème} point : Soit $G = \{y ; y = \operatorname{argmin} \{ \sum \lambda_i \| P_{C_i}(y) - y \|^2 ; y \in \mathbb{R}^n \}$ montrons alors que $F = G$.

En effet;

1) $F \supset G$. $y \in G$ donc $f(y) - f(\sum \lambda_i P_{C_i}(y)) \leq 0$ d'après (2) on a :

$$0 \leq \| \sum \lambda_i P_{C_i}(y) - y \|^2 \leq f(y) - f(\sum \lambda_i P_{C_i}(y)) \leq 0 \text{ d'où ;}$$

$$\| \sum \lambda_i P_{C_i}(y) - y \|^2 = 0 \Rightarrow y = \sum \lambda_i P_{C_i}(y) \text{ c'est à dire } y \in F$$

2) $G \supset F$. Soit $y \in F$ d'après le 2^{ème} point $y \in C$ alors $y = P_{C_i}(y) \quad \forall i$ d'où ;

$$\sum \lambda_i \| P_{C_i}(y) - y \|^2 = 0 \text{ donc } y \text{ réalise le minimum de la fonction } f \Rightarrow y \in G$$

Ainsi $F = G$.

4^{ème} point. Montrons que $\| y^{k+1} - z \|^2 \leq \| y^k - z \|^2 \quad \forall z \in C$ et $x^0 \in \mathbb{R}^n$.

$$\| y^{k+1} - z \|^2 = \| y^k - z \|^2 + (\mu^k)^2 \| \sum \lambda_i P_{C_i}(y^k) - y^k \|^2 + 2\mu^k \langle y^k - z, \sum \lambda_i P_{C_i}(y^k) - y^k \rangle$$

$$= \| y^k - z \|^2 + \mu^k (\mu^k - 2) \| \sum \lambda_i P_{C_i}(y^k) - y^k \|^2 + 2\mu^k \langle \sum \lambda_i P_{C_i}(y^k) - z, \sum \lambda_i P_{C_i}(y^k) - y^k \rangle$$

Comme $\mu^k - 2 < 0$ et $\langle \sum \lambda_i P_{C_i}(y^k) - z, \sum \lambda_i P_{C_i}(y^k) - y^k \rangle \leq 0$ alors on a :

$$\| y^{k+1} - z \|^2 \leq \| y^k - z \|^2 \quad \forall z \in C. \quad (3)$$

La suite (y^k) est féjer-monotone c'est facile de montrer qu'elle est bornée donc elle admet une sous-suite convergente vers un point $y^* \in C$, i.e.,

$$\forall \varepsilon > 0 \quad \exists j_0 \text{ tel que } \forall j > j_0 \text{ on a } \|y^{k,j} - y^*\| < \varepsilon,$$

selon (3) on a :

$$\forall m > k, j \text{ on a } \|y^m - y^*\|^2 \leq \|y^{k,j} - y^*\|^2 \text{ d'où ;}$$

$$\forall \varepsilon > 0 \quad \exists k, j \text{ tel que } \forall m > k, j \text{ on a } \|y^m - y^*\| < \varepsilon, \text{ d'où}$$

$$y^k \rightarrow y^* \in C$$

V.5.3. CAS PARTICULIER : Résolution d'un système d'inégalités.

Considérons dans \mathbb{R}^n le problème :

$$\text{"Trouver } x \in \mathbb{R}^n \text{ tel que } \langle a_i, x \rangle \leq b_i \quad i=1, \dots, m" \quad (\mathbf{T})$$

$a_i \in \mathbb{R}^n$ et b_i un élément donné de \mathbb{R} .

Algorithme : pour un $x^0 \in \mathbb{R}^n$ choisi arbitrairement on construit une suite $x^0, x^1, \dots, x^k, \dots$ par le procédé itératif suivant :

$$x^{k+1} = x^k + \mu^k \sum_{1 \leq i \leq m} \lambda_i [P_{C_i}(x^k) - x^k] \quad 0 \leq \mu^k \leq 2 \quad (1)$$

Si (\mathbf{T}) possède au moins une solution alors, d'après le théorème (3.4), l'algorithme converge vers une solution particulière de (\mathbf{T}) .

Si (\mathbf{T}) ne possède pas de solutions alors on a :

THEOREME 1 :

Sous l'hypothèse que (\mathbf{T}) ne possède pas de solutions la suite (x^k) définie par l'algorithme précédent converge vers la solution du système "aux moindres carrées" défini par :

$$f(x^*) = \min_{1 \leq i \leq m} \{1/2 \sum \lambda_i \|P_{C_i}(x) - x\|^2 \}; \quad x \in \mathbb{R}^n.$$

Preuve :

D'après le théorème précédent on a montré que $F=G$.

C'est facile de voir que :

$$F = \bigcup_{\mu^k} F_{\mu^k} \text{ où } F_{\mu^k} = \{x ; x = P_{\mu^k}(x)\} \text{ avec}$$

$$P_{\mu^k} = I + \sum \lambda_i (P_{C_i} - I) \text{ où } \sum \lambda_i = 1 \text{ et } \lambda_i > 0.$$

Montrons que $F \neq \emptyset$.

La fonction $P_{C_i}(x) - x$ est affine par morceaux. Donc f est une fonction quadratique par morceaux, et il existe un recouvrement de \mathbb{R}^n par des polyèdres fermés tel que f coïncide avec une fonction affine sur chacun d'entre eux. Comme f est minorée par 0 et en appliquant le théorème de FRANK-WOLFE [17 Corollaire] on conclut que f atteint son minimum sur chacun de ces polyèdres. Comme ils sont en nombre fini, f atteint son minimum global sur l'un d'entre eux, d'où

$$F = G \neq \emptyset$$

V.6. METHODE DU SOUS-GRADIENT POUR LA RECHERCHE DE L'INTERSECTION DE CONVEXES FERMES :

On s'intéresse toujours au problème de la recherche d'un point commun à une famille de convexes fermés $C_i, i \in I$ où I est un ensemble d'indices quelconque.

On montre que ce problème équivaut à la minimisation d'une fonction convexe non différentiable sur tout l'espace. La méthode par laquelle on résoud ce problème est celle du sous gradient.

Dans le cas fini on donne un nouvel algorithme basé sur la méthode du sous gradient qui a le même aspect que la méthode du barycentre.

V.6.1. POSITION DU PROBLEME :

Soient :

$C_i \subset \mathbb{R}^n, i \in I$ des ensembles convexes fermés de \mathbb{R}^n, I ensembles d'indices quelconques.

On désigne la distance de x à C_i par :

$$d(x, C_i) = \inf\{\|x-y\| ; y \in C_i\}$$

On suppose que $\bigcap_{1 \leq i \leq m} C_i \neq \emptyset$.

Posons $f(x) = \sup\{d^2(x, C_i) ; i \in I\}$ et $I(x) = \{i \in I ; f(x) = d^2(x, C_i)\}$.

Et ferons l'hypothèse : $(H_0) I(x) \neq \emptyset \quad \forall x \in \mathbb{R}^n$.

PROPOSITION 1 : *Trouver un point commun à la famille de convexes fermés $C_i, i \in I$, est équivalent à résoudre le problème d'optimisation convexe non différentiable suivant :*

$$(P_{6,1}) : 0 = \min\{f(x) ; x \in \mathbb{R}^n\} \text{ où } f(x) = \sup\{d^2(x, C_i) ; i \in I\}.$$

Preuve :

Si $x^* \in \bigcap_{1 \leq i \leq m} C_i$ alors $d^2(x^*, C_i) = 0 \quad \forall i \in I$, donc $f(x^*) = 0$ d'où x^* est solution du problème $(P_{6,1})$.

Soit $x^* \in \mathbb{R}^n$ tel que $0 = f(x^*) = \min\{f(x) ; x \in \mathbb{R}^n\}$ donc $\sup\{d^2(x^*, C_i) ; i \in I\} = 0$ d'où $d^2(x^*, C_i) = 0 \quad \forall i \in I$ ce qui implique $x^* \in C_i \quad \forall i \in I$ c'est à dire $x^* \in \bigcap C_i$.

On est donc en présence d'un problème d'optimisation sans contrainte où la fonction objectif f qu'on cherche à minimiser est non différentiable. L'algorithme 1 se généralise en prenant dans ce cas comme direction de déplacement à chaque étape k un sous gradient de la fonction f en x^k au lieu du gradient.

V.6.2. PRESENTATION DE LA METHODE DU SOUS-GRADIENT 1 :

Algorithme 4 :

(a) A l'étape 0 on est en x^0 (point de départ)

(b) A l'étape k on est en x^k .

Déterminer un sous-gradient $u^k \in \partial f(x^k)$.

Si $u^k=0$, c'est terminé : x^k est optimum. Sinon, définir x^{k+1} par :

$$x^{k+1} = x^k - \mu^k u^k / \|u^k\|.$$

(c) test d'arrêt; si vérifie : fin. Sinon faire $k \leftarrow k+1$ et retour en (b).

La méthode décrite ci-dessus est connue sous le nom "méthode du sous-gradient " aux **U.S.A** et "méthode de gradient généralisé" en **Union Soviétique** .

V.6.2.1 ETUDE DE LA CONVERGENCE : Le théorème classique qui étudie la convergence de cette méthode est dû à **Polyak [25]** (1967). Ce théorème donne des conditions très générales sur le choix du pas μ^k .

THEOREME 1: Soit f une fonction convexe non différentiable. Sous l'hypothèse $\sum \mu^k = +\infty$ et $\mu^k \rightarrow 0$. La suite construite par la relation

$$x^{k+1} = x^k - \mu^k u^k / \|u^k\|$$

contient une sous suite x^{n_k} telle que

$$\lim_{k \rightarrow +\infty} f(x^{n_k}) = f(x^*) = \min\{f(x) ; x \in \mathbb{R}^n\}$$

$$k \rightarrow +\infty$$

Preuve : voir **M.Michalopoulos [19]** .

Cette méthode s'appelle la méthode de la série divergente. Il y en d'autres méthodes qui diffère de celle-ci par le choix du pas de déplacements qui sont :

La méthode à pas constant : $\mu^k = \mu = \text{cste}$.

La méthode de relaxation : $\mu^k = \rho(f(x^k) - f^*) / \|u^k\|$ (voir *B.Polajk[26]*)

Où f^* est une estimation de la valeur optimale $f(x^*)$, et ρ un coefficient inférieur ou égal à 2 (coefficient de relaxation).

Revenons à notre problème $(P_{6,1})$ et lui appliquons la méthode du sous-gradient :

$$(P_{6,1}) : \min[\sup\{d^2(x, C_i) ; i \in I\} ; x \in \mathbb{R}^n]$$

On Remarque tout d'abord que pour tout $x \in C = \bigcap_{1 \leq i \leq m} C_i$ l'ensemble $S(x)$ défini par :

$S(x) = \{ y ; y = (x - P_{C_i}(x)) / \|x - P_{C_i}(x)\| \quad i \in I(x) \}$ est inclus dans $\partial f(x)$.

Ainsi l'algorithme 3 s'écrit :

pour x^0 arbitraire on a :

$$x^{k+1} = x^k + \mu^k \frac{(P_{C_{i,k}}(x^k) - x^k)}{\|P_{C_{i,k}}(x^k) - x^k\|} \quad (**)$$

$i, k \in I(x^k)$ et $\mu^k \in [\varepsilon_1, 2 - \varepsilon_2]$ où $\varepsilon_1 > 0$ et $\varepsilon_2 > 0$.

Si on considère le choix du paramètre μ^k comme suit:

$$\mu^k = \rho \frac{f(x^k) - f^*}{\|P_{C_{i,k}}(x^k) - x^k\|}$$

avec $f^* = f(x^*)$ où $x^* = \operatorname{argmin}\{f(x) ; x \in \mathbb{R}^n\}$ et $0 < \rho < 2$.

Compte-tenu du fait que $f^* = 0$ la relation $(**)$ devient :

$$x^{k+1} = x^k - \rho \frac{f(x^k)(x^k - P_{C_{i,k}}(x^k))}{\|x^k - P_{C_{i,k}}(x^k)\|^2} \quad (**)$$

Substituons la valeur de $f(x^k) = 1/2 \|P_{C_{i,k}}(x^k) - x^k\|^2$ dans $(**)$, il vient :

$$x^{k+1} = x^k + \rho (P_{C_{i,k}}(x^k) - x^k),$$

avec $0 < \varepsilon_1 \leq \rho \leq 2 - \varepsilon_2$ $\varepsilon_2 > 0$ et $i, k \in I(x^k)$.

THEOREME DE CONVERGENCE 2 : En prenant $\rho=1$ la suite (x^k) générée par l'algorithme précédent converge vers un point $x^* \in C$.

Preuve :

Soit K un convexe fermé, on définit la projection orthogonale de x sur K par $P_K(x)$, on a pour tout $x' \in K$:

$$\|x-x'\|^2 \geq \|x-P_K(x)\|^2 + \|x'-P_K(x)\|^2 \quad (1)$$

La suite (x^k) générée par l'algorithme est définie ainsi :

$$x^{k+1} = P_{C_{i,k}}(x^k) \quad \text{avec } i,k \in I(x^k).$$

D'après la propriété (5) on a $\forall x' \in C_{i,k}$, et en particulier pour tout $x' \in C$:

$$0 \leq f(x^k) = \|x^k - P_{C_{i,k}}(x^k)\|^2 \leq \|x^k - x'\|^2 - \|x' - P_{C_{i,k}}(x^k)\|^2 = \|x^k - x'\|^2 - \|x' - x^{k+1}\|^2$$

d'où pour tout $x' \in C$ on a :

$$0 \leq \|x^{k+1} - x'\|^2 \leq \|x^k - x'\|^2 \leq \|x^0 - x'\|^2 \quad (2)$$

C'est facile de voir que pour tout $k \geq 0$ on a :

$$\|x^k\| \leq \|x^k - x'\| + \|x'\| \leq \|x^0 - x'\| + \|x'\|$$

Cela traduit le fait que la suite (x^k) est bornée, donc admet au moins une valeur d'adhérence dans C .

Soit $x^* \in C$ une valeur d'adhérence de la suite (x^k) ce qui signifie que :

$$\forall \varepsilon > 0, \exists j_0 \text{ tel que } \forall j \geq j_0 \text{ on a } \|x^{k,j} - x^*\| < \varepsilon,$$

selon la relation (2) on a :

$$\forall m \geq k, j_0 \text{ on a } \|x^m - x^*\| \leq \|x^{k,j} - x^*\| < \varepsilon, \text{ d'où}$$

$$\forall \varepsilon > 0, \exists k, j_0 \text{ tel que } \forall m \geq k, j_0 \text{ on a } \|x^m - x^*\| < \varepsilon$$

D'où la fait que la suite (x^k) converge toute entière vers $x^* \in C$.

V.6.3. METHODE DU BARYCENTRE MODIFIEE :

Dans cette partie, on développe une méthode pour la recherche d'un point commun à une famille de convexes fermés basée sur la méthode du sous-gradient. Elle diffère de la méthode du barycentre par le fait que cette dernière consiste à projeter à chaque itération un même point sur tous les convexes puis à prendre leurs barycentres, tandis que notre méthode consiste à projeter le même point sur les convexes les plus éloignés puis à prendre leurs barycentres.

Dans le cas où l'intérieur de cette intersection est non vide, l'algorithme proposé converge en un nombre fini d'itérations.

V.6.3.1. POSITION DU PROBLEME :

Soit le problème :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } x \in \bigcap_{1 \leq i \leq m} C_i, \quad (P_{6,1})$$

$C_i \subset \mathbb{R}^n \quad i=1, \dots, m$ des convexes fermés.

On désigne la distance de x à C_i par : $d(x, C_i) = \|x - P_{C_i}(x)\|$

où P_{C_i} est la projection orthogonale sur C_i .

Remarque : Dans la pratique on est amené toujours à considérer des convexes où la projection est facile à mettre en œuvre tels que des demi-espaces, des hyperplans et des sphères fermées euclidiennes.

Le problème $(P_{6,1})$ est équivalent à :

$$0 = \min\{f(x) ; x \in \mathbb{R}^n\}, \quad (P_{6,2})$$

avec $f(x) = \sup\{1/2d^2(x, C_i) ; 1 \leq i \leq m\} = \sup\{1/2\|x - P_{C_i}(x)\|^2 ; 1 \leq i \leq m\}$,

posons $I(x) = \{1 \leq i \leq m ; f(x) = 1/2d^2(x, C_i)\}$, et ferons l'hypothèse que $I(x) \neq \emptyset$ pour tout x .

Du fait que la fonction $f(\cdot)$ est non-différentiable la méthode par laquelle on résout le problème $(P_{6,2})$ est celle du sous-gradient qui consiste à construire à partir d'un point $x^0 \in \mathbb{R}^n$ une suite $x^0, x^1, \dots, x^k, \dots$ par le processus itératif suivant:

$$x^{k+1} = x^k - \left(\frac{u^k}{\|u^k\|} \right) \quad \text{où } u^k \in \partial f(x^k)$$

Or dans ce cas le sous-différentiel de $f(\cdot)$ au point x^k est donné par la relation suivante:

$$\partial f(x^k) = \sum_{i \in I(x^k)} \lambda_i \partial f_i(x^k) \quad \text{avec} \quad \sum_{i \in I(x^k)} \lambda_i = 1 \quad \text{et} \quad \lambda_i > 0.$$

Ainsi, la procédure pour déterminer un point commun à l'intersection de la famille C_i pour $i=1, \dots, m$ se formule comme suit :

pour $x^0 \in \mathbb{R}^n$ choisi arbitrairement, l'itéré x^{k+1} pour $k=0, 1, \dots$, est donné par :

$$x^{k+1} = x^k + \mu^k \frac{\sum_{i \in I(x^k)} (P_{C_i}(x^k) - x^k)}{\| \sum_{i \in I(x^k)} (P_{C_i}(x^k) - x^k) \|} \quad \sum_{i \in I(x^k)} \lambda_i = 1 \quad \text{et} \quad \lambda_i > 0$$

Selon le choix du paramètre μ^k pour tout $k \geq 0$ on a une des méthodes connues en littérature.

Nous présentons maintenant un théorème de convergence finie pour le cas où les μ^k sont données par la série divergente i.e., μ^k vérifiée :

$$\mu^k \rightarrow 0 \quad \text{et} \quad \sum_{i=1}^{\infty} \mu^k = +\infty$$

THEOREME 1 Polyak [25] : Soient $(C_i)_{1 \leq i \leq m}$ une famille de convexes fermés. Si on a a) $\text{Int}(\cap C_i) \neq \emptyset$.

b) $\mu^k \rightarrow 0$ et $\sum_{i=1}^{\infty} \mu^k = +\infty$

Alors la méthode présentée ci-dessus converge en un nombre fini d'itérations, i.e., il existe un k tel que $x^k \in (\cap C_i)$.

V.6.4. ALGORITHME DU SOUS-GRADIENT 2 POUR LE CALCUL DE L'INTERSECTION DE DEUX CONVEXES DANS L'ESPACE PRODUIT :

Nous proposons un nouvel algorithme basé sur la méthode du sous-gradient, destinée à minimiser une fonction convexe non différentiable, pour la recherche d'un point commun à une famille de convexes fermés. On constate que cet algorithme est identique à l'algorithme 1.

Soit le problème :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } x \in \bigcap_{1 \leq i \leq m} C_i \quad (\mathbf{P}_{6,3}).$$

Soit $\mathcal{C} = \bigcap C_i$, \mathcal{D} le convexe diagonal et $\mathcal{X} = \mathbb{R}^n \times \dots \times \mathbb{R}^n$ (m fois), l'espace ainsi défini est doté de la topologie produit par le produit scalaire $((\cdot, \cdot)) = \sum \lambda_i (\cdot, \cdot)$, la norme qui lui est associée notée $\|\cdot\|$ et la distance $d((\cdot, \cdot))$.

Le problème $(\mathbf{P}_{6,3})$ est équivalent au problème suivant :

$$(\mathbf{P}_{6,4}) : 0 = \min\{f(\mathcal{X}) ; \mathcal{X} \in \mathcal{X}\} \text{ où } f(\mathcal{X}) = \text{Sup}\{d^2((\mathcal{X}, \mathcal{C})); d^2((\mathcal{X}, \mathcal{D}))\},$$

en effet, si $x^* \in \bigcap C_i$ alors $\mathcal{X}^* = (x^*, \dots, x^*) \in \mathcal{C} \cap \mathcal{D}$ d'où

$$d^2((\mathcal{X}^*, \mathcal{C})) = d^2((\mathcal{X}^*, \mathcal{D})) = 0 \Rightarrow f(\mathcal{X}^*) = 0 \text{ donc } \mathcal{X}^* \text{ est solution du problème } (\mathbf{P}_{6,4}).$$

Si \mathcal{X}^* est solution du problème $(\mathbf{P}_{6,3})$ alors $f(\mathcal{X}^*) = 0$ i.e., $d^2((\mathcal{X}^*, \mathcal{C})) = 0$ et $d^2((\mathcal{X}^*, \mathcal{D})) = 0$ d'où $\mathcal{X}^* = (x^*, x^*, \dots, x^*) \in \mathcal{C} \cap \mathcal{D} \Rightarrow x^* \in \bigcap C_i$.

On propose pour résoudre $(\mathbf{P}_{6,3})$ un nouvel algorithme basé sur la méthode du sous-gradient. On constate que le processus itératif de cette nouvelle méthode est le même que celui donné par *l'algorithme 3*.

Algorithme 4 :

Soit $x^0 = (x^0, x^0, \dots, x^0) \in \mathcal{X}$ choisi arbitrairement.

Supposons qu'à la $k^{\text{ième}}$ itération on a $x^k \in \mathcal{D}$.

On calcule y^{k+1} par la relation suivante :

$$y^{k+1} = x^k - \mu^k [x^k - P_{\mathcal{C}}(x^k)]$$

$$\mu^k \in [\varepsilon_1, 2 - \varepsilon_2] \text{ où } y^{k+1} = (y_1^{k+1}, \dots, y_m^{k+1}).$$

D'après la proposition la projection de x^k sur \mathcal{C} est donné par :

$$P_{\mathcal{C}}(x^k) = (P_{C_1}(x^k), P_{C_2}(x^k), \dots, P_{C_m}(x^k)),$$

d'où $y_i^{k+1} = x^k - \mu^k [x^k - P_{C_i}(x^k)].$

Projetons le vecteur y^{k+1} sur \mathcal{D} on aura :

$$x^{k+1} = (x^{k+1}, \dots, x^{k+1}) = P_{\mathcal{D}}(y^{k+1}).$$

$$\text{Or } P_{\mathcal{D}}(y^{k+1}) = \left(\sum_{i=1}^m \lambda_i y_i^{k+1}, \dots, \sum_{i=1}^m \lambda_i y_i^{k+1} \right)$$

$$\text{D'où } x^{k+1} = \sum_{i=1}^m \lambda_i y_i^{k+1} = \sum_{i=1}^m \lambda_i (x^k - \mu^k (x^k - P_{C_i}(x^k)))$$

$$\text{Ou bien } x^{k+1} = x^k + \mu^k \sum_{i=1}^m \lambda_i [P_{C_i}(x^k) - x^k] \quad \varepsilon_1 < \mu^k < 2 - \varepsilon_2 \quad \varepsilon_1 > 0 \text{ et } \varepsilon_2 > 0$$

En résumé *l'algorithme 4* se décrit comme suit :

$x^0 \in \mathbb{R}^n$ un point choisi arbitrairement.

Alors le processus itératif défini par cet algorithme est comme suit:

$$x^{k+1} = x^k + \mu^k \sum_{i=1}^m \lambda_i [P_{C_i}(x^k) - x^k] \quad 0 < \varepsilon_1 < \mu^k < 2 - \varepsilon_2 \text{ et } \varepsilon_2 > 0 .$$

V.7. ALGORITHME DU MINIMAX POUR LA RECHERCHE D'UN POINT APPARTENANT A L'INTERSECTION DE CONVEXES FERMES :

Dans cette partie on présente une nouvelle méthode, pour la recherche d'un point commun à une famille quelconque de convexes fermés, basée sur la théorie du minimax (voir B.MARTINET[17]). On constatera du coup qu'elle est équivalente à la méthode des projections successives.

V.7.1. POSITION DU PROBLEME :

Soit le problème :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } x \in \bigcap_{i \in I} C_i \quad (\mathbf{P}_{7,1}).$$

Où I est un ensemble d'indices quelconques.

D'après l'étude faite au paragraphe précédent le problème $(\mathbf{P}_{7,1})$ est équivalent à :

$$(\mathbf{P}_{7,1}) : 0 = \alpha = \min\{(\sup(f_i(x) ; i \in I)) ; x \in \mathbb{R}^n\},$$

$$\text{où } f_i(x) = 1/2 \|x - P_{C_i}(x)\|^2.$$

Selon la théorie de la dualité du minimax, le problème dual de $(\mathbf{P}_{7,1})$ s'écrit comme suit :

$$(\mathbf{Q}_{7,1}) : 0 = \beta = \sup\{(\min(f_i(x) ; x \in \mathbb{R}^n)) ; i \in I\}$$

Soit (\mathcal{P}) l'ensemble de solutions du problème $(\mathbf{P}_{7,1})$ et (\mathcal{Q}) l'ensemble de solution du problème $(\mathbf{Q}_{7,1})$.

DEFINITION 1: On dit qu'un couple $(i^*, x^*) \in I \times \mathbb{R}^n$ est un point selle de $f_i(x)$ sur $I \times \mathbb{R}^n$ si :

$$f_i(x^*) \leq f_{i^*}(x^*) \leq f_{i^*}(x) \quad \forall x \in \mathbb{R}^n \text{ et } \forall i \in I$$

PROPOSITION 1 : La fonction $f_i(x)$ définie sur $I \times \mathbb{R}^n$ à valeurs réelles possède un point selle (i^*, x^*) sur $I \times \mathbb{R}^n$ si et seulement si :

$$0 = \min\{(\sup(f_i(x) ; i \in I)) ; x \in \mathbb{R}^n\} = \sup\{(\min(f_i(x) ; x \in \mathbb{R}^n)) ; i \in I\} = f_{i^*}(x^*)$$

REMARQUE : Si (i^*, x^*) est un point selle alors $i^* \in \mathcal{Q}$ et $x^* \in \mathcal{P}$.

Le problème primal est :

$$(P_{7,1}) : 0 = \alpha = \min\{(\sup_{i \in I} f_i(x) ; x \in \mathbb{R}^n)\},$$

et le problème dual est :

$$(Q_{7,1}) : 0 = \beta = \sup\{(\min_{x \in \mathbb{R}^n} f_i(x) ; i \in I)\}.$$

On a de plus $\alpha = \beta = 0$ donc d'après la proposition précédente il existe un point selle $(i^*, x^*) \in I \times \mathbb{R}^n$ de $f_i(x)$ où $i^* \in I$ est solution du problème $(Q_{7,1})$ et $x^* \in \mathbb{R}^n$ est solution du problème $(P_{7,1})$.

Il est facile de voir que la résolution du problème $(Q_{7,1})$ est moins difficile que la résolution du problème $(P_{7,1})$.

Nous proposons pour résoudre $(Q_{7,1})$ un algorithme dont sa description est la suivante :

Algorithme 5 :

Initialisation : Soit $x^0 \in \mathbb{R}^n$ un point arbitrairement choisi.

Etape principale : A l'itération k on a x^k ,

a) On détermine i,k comme suit :

$$i,k \in \arg(\sup\{f_i(x^k) ; i \in I\})$$

b) Une fois i,k est donné on calcule x^{k+1} par :

$$x^{k+1} \in \arg(\min\{f_{i,k}(x) ; x \in \mathbb{R}^n\})$$

Si nous explicitons les étapes de l'algorithme on aura :

à l'étape a) on a déterminer i,k tel que $i,k \in \arg(\sup\{f_i(x^k) ; x \in \mathbb{R}^n\})$ avec

$$f_i(x^k) = 1/2 \|x^k - P_{C_i}(x^k)\|^2.$$

Posons $I(x) = \{i \in I ; f_i(x) = \sup\{f_i(x) ; i \in I\}\}$

$i,k \in \arg(\sup\{f_i(x^k) ; i \in I\}) \Leftrightarrow i,k \in I(x^k) \Leftrightarrow i,k$ est l'indice du convexe le plus éloigné par rapport à x^k .

A l'étape b) on calcule x^{k+1} par

$$x^{k+1} \in \arg(\min\{f_{i,k}(x) = 1/2 \|x - P_{C_{i,k}}(x)\|^2 ; x \in \mathbb{R}^n\})$$

x^{k+1} est le point qui réalise le minimum de la fonction $f_{i,k}(x) = 1/2 \|x - P_{C_{i,k}}(x)\|^2$ sur \mathbb{R}^n , or le minimum est atteint pour un point appartenant à $C_{i,k}$. Donc $x^{k+1} = P_{C_{i,k}}(x^k)$.

Ainsi l'algorithme génère une suite (x^k) de \mathbb{R}^n par le processus itératif, en prenant à l'initialisation un point $x^0 \in \mathbb{R}^n$ choisi arbitrairement, suivant:

$$x^{k+1} = P_{C_{i,k}}(x^k) \quad i, k \in I(x^k) = \{j \in I ; 1/2 \|x^k - P_{C_j}(x^k)\|^2 = \sup\{1/2 \|x^k - P_{C_i}(x^k)\|^2 ; i \in I\}$$

Dans le cas de deux convexes on reconnaît la méthode des projections successives décrite au paragraphe (1.4), et dans le cas de plusieurs convexes on reconnaît la méthode décrite à la page 153 avec $\rho=1$.

V.8. RESUME GENERAL :

Reprenons le problème de la recherche d'un point appartenant à l'intersection de convexes fermés

$$(P_{8,1}) : \text{Trouver } x \in \mathbb{R}^n \text{ tel que } x \in \bigcap_{1 \leq i \leq m} C_i$$

En un premier temps, on a montré que ce problème est équivalent, dans l'espace produit, au problème suivant :

$$\min\{1/2 \|x - P_{\mathcal{D}}(x)\|^2 ; x \in \mathcal{D}\},$$

la méthode par laquelle on a résolu ce problème est celle du gradient projeté dont le processus itératif est le suivant :

$$x^{k+1} = x^k + \tau^k [P_{\mathcal{D}}(P_{\mathcal{G}}(x^k)) - x^k],$$

en revenant aux espaces facteurs \mathbb{R}^n le processus itératif prend la forme :

$$x^{k+1} = x^k + \mu^k \sum_{1 \leq i \leq m} \lambda_i [P_{C_i}(x^k) - x^k] \quad \text{avec } \sum_{1 \leq i \leq m} \lambda_i = 1 \text{ et } \lambda_i > 0 \text{ et } \mu^k > 0$$

On a montré de plus que la suite (x^k) générée par ce processus converge vers un point commun à la famille des convexes $(C_i)_{1 \leq i \leq m}$, i.e., la résolution du problème $(P_{8,1})$.

En un deuxième lieu, on a montré que le problème $(P_{8,1})$ est équivalent au problème suivant :

$$\min\{1/2\sum_{1 \leq i \leq m} \lambda_i \|x - P_{C_i}(x)\|^2 ; x \in \mathbb{R}^n\}$$

C'est un problème d'optimisation convexe sans contrainte qu'on a résolu par la méthode du gradient dont le procédé est le suivant:

x^0 quelconque de \mathbb{R}^n , le processus donne

$$x^{k+1} = x^k + \mu^k \sum_{1 \leq i \leq m} \lambda_i (P_{C_i}(x^k) - x^k) \quad \text{avec } \sum_{1 \leq i \leq m} \lambda_i = 1, \lambda_i > 0 \text{ et } \mu^k > 0.$$

En un dernier temps, on a montré que le problème $(P_{8,1})$ est équivalent au problème :

$$\min\{\sup[1/2d^2(x, \mathcal{C}), 1/2d^2(x, \mathcal{D})] ; x \in \mathcal{E}\}$$

qu'on a résolu par la méthode du sous-gradient qui est conforme à la procédure suivante :

$$x^{k+1} = P_{\mathcal{D}}[x^k + \mu^k (P_{\mathcal{E}}(x^k) - x^k)]$$

Comme l'opérateur $P_{\mathcal{D}}$ est linéaire alors on a :

$$x^{k+1} = x^k + \mu^k [P_{\mathcal{D}} P_{\mathcal{E}}(x^k) - x^k]$$

En revenons à l'espace \mathbb{R}^n on a :

$$x^{k+1} = x^k + \mu^k \sum_{1 \leq i \leq m} \lambda_i [P_{C_i}(x^k) - x^k]$$

On conclut que ces trois méthodes pour la résolution de trois problèmes distincts engendrent, pour un choix de pas convenable, le même algorithme pour résoudre le problème $(P_{8,1})$ dont le processus itératif est comme suit :

$$x^{k+1} = x^k + \mu^k \sum_{1 \leq i \leq m} \lambda_i [P_{C_i}(x^k) - x^k] \quad \text{avec } \sum_{1 \leq i \leq m} \lambda_i = 1, \lambda_i > 0 \text{ et } \mu^k > 0 \forall k \geq 0$$

V.9. RECHERCHE DE L'INTERSECTION D'UNE FAMILLE DE CONVEXES PAR CROISEMENT DE METHODES DE DESCENTE :

A travers cette partie, on présente une généralisation de l'algorithme de croisement de méthodes de descente pour la recherche de l'intersection-ou de la distance dans le cas où l'intersection est vide- de deux convexes due à G.PIERRA[22] au cas de plusieurs convexes.

V.9.1. POSITION DU PROBLEME :

Soit le problème :

$$(P_{9,1}) : \text{Trouver } x \in \mathbb{R}^n \text{ tel que } x \in \bigcap_{1 \leq i \leq m} C_i,$$

pour $i=1, \dots, m$, $C_i \subset \mathbb{R}^n$ des convexes fermés.

On considère le cas général où on connaît pas les projecteurs des convexes C_i et que ces derniers ne sont obtenus que par une méthode algorithmique.

V.9.2. NOUVELLE FORMULATION DU PROBLEME :

On introduit :

$\mathcal{H} = \mathbb{R}^n \times \mathbb{R}^n \times \dots \times \mathbb{R}^n$ doté de la topologie produit par le produit scalaire $((\dots)) = \sum_{1 \leq i \leq m} (\dots)$ et de la norme associée notée $||| \cdot |||$.

On désigne l'élément de \mathcal{H} par $\mathcal{X} = (x_1, x_2, \dots, x_m)$ avec $x_i \in \mathbb{R}^n$.

Posons $\mathcal{C} = C_1 \times C_2 \times \dots \times C_m$ et le convexe diagonal :

$$\mathcal{D} = \{ \mathcal{Y} \in \mathcal{H} ; \mathcal{Y} = (y, y, \dots, y) ; y \in \mathbb{R}^n \}.$$

Nous rappelons ici quelques propriétés établies au paragraphe :

Le problème $(P_{9,1})$ équivaut, dans l'espace produit, au problème suivant :

Trouver $(\mathcal{X}^*, \mathcal{Y}^*) \in \mathcal{C} \times \mathcal{D}$ tel que

$$1/2 ||| \mathcal{X}^* - \mathcal{Y}^* |||^2 = \min \{ 1/2 ||| \mathcal{X} - \mathcal{Y} |||^2 ; (\mathcal{X}, \mathcal{Y}) \in \mathcal{C} \times \mathcal{D} \} \quad (P_{9,2})$$

Qui peut s'écrire sous la forme :

Trouver $(x_1^*, x_2^*, \dots, x_m^*, y^*) \in C = C_1 \times C_2 \times \dots \times C_m \times \mathbb{R}^n$ tel que :

$$\min_{1 \leq i \leq m} \lambda_i \|x_i^* - y^*\|^2 = \min_{1 \leq i \leq m} \{1/2 \sum \lambda_i \|x_i - y\|^2 ; (x_1, \dots, x_m, y) \in C\}, \quad (P_{9,3})$$

posons $f(x_1, x_2, \dots, x_m, y) = 1/2 \sum_{1 \leq i \leq m} \lambda_i \|x_i - y\|^2$. $f(\cdot)$ est une fonction à valeurs réelles

définie sur $\mathbb{R}^n \times \mathbb{R}^n \times \dots \times \mathbb{R}^n$ ($m+1$ fois).

La recherche d'un point commun à une famille de convexes fermés revient donc à résoudre le problème d'optimisation, dans l'espace produit \mathcal{H} , qui s'écrit de la manière suivante :

$$(P_{9,3}) \quad \text{Trouver } u^* \in C \text{ tel que } f(u^*) = \min\{f(u) ; u \in C\},$$

où $u^* = (x_1^*, x_2^*, \dots, x_m^*, y^*)$ et $u = (x_1, \dots, x_m, y)$.

La méthode de décomposition qu'on propose pour résoudre $(P_{9,3})$ consiste à construire à partir d'un élément arbitraire $u^0 \in \mathcal{H}$ une suite u^0, u^1, \dots, u^k d'éléments de \mathcal{H} qui convergent vers une solution u^* du problème $(P_{9,3})$.

On note : $u^k = (x_1^k, x_2^k, \dots, x_m^k, y^k)$ où $x_i^k \in C_i$ pour $i=1, \dots, m$ et $y^k \in \mathbb{R}^n$,

et $f_{i,k}(x_i) = f(x_1^{k+1}, x_2^{k+1}, \dots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \dots, x_m^k, y^k)$ pour $i=1, \dots, m$,

et $f_{m+1,k}(y) = f(x_1^{k+1}, x_2^{k+1}, \dots, x_m^{k+1}, y)$.

La méthode consiste ,pour passer de u^k à u^{k+1} , à déterminer successivement pour $i=1, \dots, m$ l'élément x_i^{k+1} et pour $i=m+1$ l'élément y^{k+1} à partir de $x_1^k, x_2^k, \dots, x_m^k, y^k$ en effectuant une seule itération d'une méthode de descente M destinée à résoudre le problème suivant :

$$(P_{i,k}) \quad \min\{f_{i,k}(x_i) ; x_i \in C_i\} \quad \text{pour } i=1, \dots, m$$

et pour $i=m+1$ le problème :

$$(P_{m+1,k}) \quad \min\{f_{m+1,k}(y) ; y \in \mathbb{R}^n\}$$

Les sous-problèmes résolus lors de chacun des pas de l'algorithme de décomposition, s'écrivent ainsi:

Trouver $x_i^{k+1} \in C_i$ pour $i=1, \dots, m$ tels que

$$f_{i,k}(x_i^{k+1}) = \min\left\{\frac{1}{2} \sum_{j=1}^{i-1} \lambda_j \|x_j^{k+1} - y^k\|^2 + \frac{1}{2} \lambda_i \|x_i - y^k\|^2 + \frac{1}{2} \sum_{j=i+1}^m \lambda_j \|x_j^k - y^k\|^2 ; x_i \in C_i\right\} \quad (P_{i,k})$$

et Trouver $y^{k+1} \in \mathbb{R}^n$ tel que

$$(P_{m+1,k}) f_{m+1,k}(y^{k+1}) = \min_{1 \leq i \leq m} \{1/2 \sum \lambda_i \|x_i^{k+1} - y\|^2 ; y \in \mathbb{R}^n\},$$

au lieu de résoudre à chaque pas le problème $(P_{i,k})$; nous allons donc seulement effectuer une itération d'une méthode de descente -cf CEA[6] - à partir de x_i^k et de y^k .

Pour résoudre le problème $(P_{i,k})$ $i=1, \dots, m$, on définit à partir de x_i^k une direction de descente p_i^k par rapport au gradient :

$$g_i^k = y^k - x_i^k$$

x_i^{k+1} sera alors le point x_i de C_i de la forme :

$$x_i^k + \lambda_i p_i^k ; \lambda_i \geq 0.$$

Pour lequel $f_{i,k}(x_i)$ est minimal.

-Choix de p_i^k , pour $i=1, \dots, m$: Soit r_i donné, strictement positif, on définit C_i^k

$$\text{par : } C_i^k = \{x_i ; x_i \in C_i ; \|x_i - x_i^k\| \leq r_i\}$$

Soit alors un élément $z_i^k \in C_i$ qui vérifie la relation (1) suivante :

$$\begin{cases} \langle g_i^k, z_i^k \rangle \geq \langle g_i^k, x_i \rangle \quad \forall x_i \in C_i^k \\ \|z_i^k\| \leq L_i \end{cases}$$

avec L_i , une constante donnée.

On prend alors :

$$p_i^k = z_i^k - x_i^k \quad (2).$$

- Détermination de x_i^{k+1} pour $i=1, \dots, m$:

Posons $x_i^{k+1}(\lambda_i) = x_i^k + \lambda_i p_i^k$

On sait que : $g_i^k = y^k - x_i^k$.

$$f_{i,k}(x_i^{k+1}(\lambda_i)) = 1/2 \sum \|x_j^{k+1} - y^k\|^2 + 1/2 \|x_i^k + \lambda_i p_i^k - y^k\|^2 + 1/2 \sum \|x_i^k - y^k\|^2$$

La fonction $\lambda_i \rightarrow f_{i,k}(x_i^{k+1}(\lambda_i))$ atteint son minimum pour

$$\lambda_{i,1}^k = \langle g_i^k, p_i^k \rangle / \|p_i^k\|^2$$

Soit d'autre part :

$$\lambda_{i,2}^k = \max\{\lambda_i \geq 0 ; x_i^k + \lambda_i p_i^k \in C_i\},$$

le meilleur choix possible de λ_i est donc :

$$\lambda_i^k = \min\{\lambda_{i,1}^k, \lambda_{i,2}^k\},$$

et l'on prend:

$$x_i^{k+1} = x_i^k + \lambda_i^k p_i^k \quad \text{pour } i=1, \dots, m.$$

Une fois les points x_i^{k+1} sont déterminés, on résoud le problème $(P_{m+1,k})$ en définissant à partir de y^k la direction de descente $p_{m+1}^k = \nabla f_{i,k}(y^k)$, y^{k+1} sera alors:

$$y^{k+1} = y^k - \lambda_{m+1} \nabla f_{i,k}(y^k),$$

or $\nabla f_{i,k}(y^k) = \sum_{1 \leq i \leq m} [y^k - x_i^{k+1}]$, d'où

$$y^{k+1} = y^k + \lambda_{m+1} \sum_{1 \leq i \leq m} [x_i^{k+1} - y^k]$$

Ainsi l'algorithme est conforme à la procédure suivante :

partant de $x_1^0 \in C_1, \dots, x_m^0 \in C_m$ et $y^0 \in \mathbb{R}^n$ choisis arbitrairement. Le passage de x_1^k, \dots, x_m^k, y^k à $x_1^{k+1}, \dots, x_m^{k+1}$ s'effectue de la façon suivante :

on pose

$$g_i^k = y^k - x_i^k,$$

et on suppose donnés les nombres r_i et L_i , on choisit une direction de descente p_i^k par (2).

Le point x_i^{k+1} pour $i=1, \dots, m$ est alors le point qui minimise $f_{i,k}(x_i)$ sur le segment :

$$\{x_i + \lambda_i p_i^k ; \lambda_i \geq 0\} \cap C_i$$

C'est à dire :

$$x_i^{k+1} = x_i^k + \lambda_i^k p_i^k$$

Une fois les x_i^{k+1} sont déterminés, le passage de x_1^k, \dots, x_m^k, y^k à y^{k+1} est donné par :

$$y^{k+1} = y^k + \lambda_{m+1}^k \sum_{i=1}^m (x_i^{k+1} - y^k) \quad \text{où } \lambda_{m+1}^k > 0.$$

THEOREME 1 : *Sous l'hypothèse (H) la suite $\{u^k\}$ générée par l'algorithme précédent est minimisante pour la fonction $f(u)$, et toute valeur d'adhérence de cette suite -il en existe au moins une- est solution du problème (P_{9,3}).*

Preuve : Supposons que la suite $\{u^k\}$ infinie. On a, par construction, pour tout k :

$$f(u^k) \geq f_{i,k}(x_i^{k+1}) \geq \dots \geq f_{i,k}(x_i^{k+1}) = f_{i+1,k}(x_i^k) \geq f_{i+1,k}(x_i^{k+1}) \geq f(u^{k+1})$$

Puisque la suite $\{f(u^k)\}$ est bornée inférieurement, donc elle converge, et on a pour tout k et tout $i=1, \dots, m$:

$$\lim_{k \rightarrow \infty} [f_{i,k}(x_i^{k+1}) - f_{i,k}(x_i^k)] = 0 \quad (***)$$

D'autre part, puisque $f_{i,k}(x_i)$ est fortement convexe pour $i=1, \dots, m$ et tout k ,

l'ensemble : $S_{i,k} = \{x_i \in \mathbb{R}^n ; f_{i,k}(x_i) \leq f_{i,k}(x_0)\}$ est borné.

Comme $x_i^k \in S_{i,k}$ pour tout k alors la suite x_i^k est bornée, i.e., $\|x_i^k\| \leq c_i$ pour tout $i=1, \dots, m$.

De plus $\|z_i^k\|$ est borné, et du fait que

$$\|p_i^k\| = \|z_i^k - x_i^k\| \leq \|z_i^k\| + \|x_i^k\| \leq M_i \quad \forall i$$

alors $\|p_i^k\|$ est borné. En plus on a :

$$f_{i,k}(x_i^k) - f_{i,k}(x_i^{k+1}) = 1/2 \frac{\langle g_i^k, p_i^k \rangle}{\|p_i^k\|^2}$$

On en déduit d'après (***) que :

$$\lim_{k \rightarrow \infty} \langle g_i^k, p_i^k \rangle = 0,$$

d'après (1) on a :

$$\langle g_i^k, z_i^k \rangle \geq \langle g_i^k, x_i \rangle \quad \text{pour tout } x_i \in C_i^k,$$

ce qui est équivalent à :

$$\langle g_i^k, p_i^k + x_i^k \rangle \geq \langle g_i^k, x_i \rangle \quad \text{pour tout } x_i \in C_i^k,$$

ou bien :

$$\langle g_i^k, p_i^k \rangle \geq \langle g_i^k, x_i - x_i^k \rangle \quad \text{pour tout } x_i \in C_i^k$$

Comme on peut considérer C_i comme borné on aura alors :

$$\limsup \langle g_i^k, x_i - x_i^k \rangle \leq 0 \quad \text{pour tout } x_i \in C_i^k,$$

or $g_i^k = -\nabla f_{i,k}(x_i^k)$ Comme on peut considérer C_i comme borné il vient :

$$\limsup [\langle -\nabla f_{i,k}(x_i^k), x_i - x_i^k \rangle] \leq 0 \quad \text{pour tout } x_i \in C_i,$$

ou bien

$$\liminf [\langle \nabla f_{i,k}(x_i^k), x_i - x_i^k \rangle] \geq 0 \quad \text{pour tout } x_i \in C_i,$$

la forte convexité de f par rapport à chaque variable, s'exprime par :

$$f_{i,k}(x_i^{k+1}) - f_{i,k}(x_i^k) = \langle \nabla f_{i,k}(x_i^k), x_i^{k+1} - x_i^k \rangle + K_i \|x_i^{k+1} - x_i^k\|^2,$$

d'où

$$\lim_{k \rightarrow \infty} \|x_i^{k+1} - x_i^k\| = 0 \text{ pour } i=1, \dots, m.$$

Comme la dérivée de f étant Lipschitzienne, il vient:

$$\liminf ((\nabla f(u^k), u - u^k)) \geq 0 \text{ pour tout } u \in C,$$

or la fonction f est convexe, ceci entraîne :

$$f(u) \geq f(u^k) + (\nabla f(u^k), u - u^k) \text{ pour tout } u \in C,$$

d'où en passant à la limite dans le second membre :

$$f(u) \geq \liminf_{k \rightarrow \infty} f(u^k) \text{ pour tout } u \in C,$$

et la suite (u^k) est minimisante pour f .

De plus, la suite $\{u^k\}$ étant bornée, elle possède sur C convexe fermé, au moins un point d'adhérence faible, qui est la solution du problème $(P_{9,1})$.

V.10. UN SURVOL DES METHODES DE PROJECTIONS :

On distingue deux classes de méthodes de projection pour la recherche d'un point commun à une famille de convexes fermés. La première englobe les méthodes qui ne font intervenir à chaque itération qu'un seul convexes. Elles sont connues sous le nom de méthodes de projections successives. La plus ancienne de ces méthodes est celle de S.KACZMARZ [12] qui consiste à résoudre un système de grande taille.

La deuxième classe est constituée des méthodes de type barycentrique et qui consistent à projeter simultanément sur tous les convexes puis à prendre leur barycentre. La plus célèbre méthode de cette classe est le procédé de G.CIMMINO[12] qui consiste à résoudre un système d'équations linéaires.

Nous présentons ces méthodes dans un cadre théorique général en commençant par la première classe.

V.10.1 METHODE DE PROJECTIONS SUCCESSIVES :

V.10.1.1. METHODE DE KACZMARZ :

Cette méthode consiste à résoudre un système d'équations à grande taille en utilisant la projection des itérés sur les hyperplans correspondants aux systèmes.

Le problème est de ce type :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } \langle a_i, x \rangle - b_i = 0 \text{ pour } i=1, \dots, m,$$

où a_i est un élément de \mathbb{R}^n et $b_i \in \mathbb{R}$. On suppose que ce problème admet une solution.

La méthode de KACZMARZ[12] consiste à construire une suite $x^0, x^1, \dots, x^k, \dots$ avec x^0 comme point de départ, par la procédure suivante :

$$x^{k+1} = x^k + \lambda^k [(b_{i,k} - \langle a_{i,k}, x^k \rangle) / \|a_{i,k}\|^2] a_{i,k}$$

Où $i, k = k+1 [\text{mod } m]$ (i.e., i, k représente le reste de la division de $k+1$ par m).

Et le paramètre de relaxation λ^k est tel que $0 < \epsilon \leq \lambda^k \leq 2 - \epsilon_2$, $\epsilon_2 > 0$

9.1.1.1) INTERPRETATION GEOMETRIQUE :

Etant donné x^k et l'hyperplan $C_{i,k} = \{x \in \mathbb{R}^n ; \langle a_{i,k}, x \rangle - b_{i,k} = 0\}$, déterminé par la i, k équation, x^{k+1} appartient à la droite passant par x^k et perpendiculaire à $C_{i,k}$. Si le paramètre de relaxation est égal à 1 ($\lambda^k = 1$) pour $k \geq 0$, x^{k+1} est la projection orthogonale de x^k sur $C_{i,k}$.

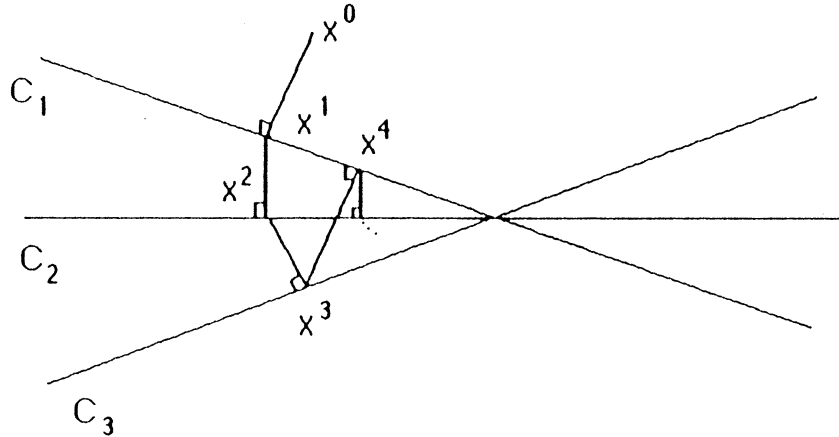


Fig 1. La méthode de Kaczmarz avec $\lambda^k=1$.

CONVERGENCE DE LA SUITE : Les itérés produits par la méthode de KACZMARZ[12], dans le cas où le système d'équations est consistant, converge vers la solution x^* du système de norme minimale à condition que $x^0 \in R(A')$ l'image de A' . Pour plus de détails voir N.GASTINEL[12].

V.10.1.2. METHODE DE RELAXATION DE AGMON, MOTZKIN ET SHCENBERG :

Cette méthode est une généralisation du procédé de KACZMARZ pour la résolution d'un système d'inégalités linéaires [1] et [20].

Le problème est :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } \langle a_i, x \rangle - b_i \leq 0 \text{ pour } i=1, \dots, m$$

PRESENTATION DE LA METHODE :

La méthode consiste, à partir d'un point x^0 choisi arbitrairement à construire une suite $x^0, x^1, \dots, x^k, \dots$ selon le processus itératif suivant :

$$x^{k+1} = x^k + c^k a_{i,k} \quad \text{où} \quad c^k = \min\{0, \lambda^k (b_{i,k} - \langle a_{i,k}, x^k \rangle) / \|a_{i,k}\|^2\} \text{ avec}$$

$0 < \varepsilon_1 \leq \lambda^k \leq 2 - \varepsilon_2$, $\varepsilon_2 > 0$, et $i^k = k+1 \pmod{m}$ (i.e., i, k représente le reste de la division de $k+1$ par m).

INTERPRETATION GEOMETRIQUE :

Etant donné x^k et un demi-espace fermé $C_{i,k} = \{x \in \mathbb{R}^n ; \langle a_{i,k}, x \rangle - b_{i,k} \leq 0\}$, déterminé par la i^k ème inégalité, si $x^k \in C_{i,k}$ alors $x^{k+1} = x^k$. Sinon i.e., $x^k \notin C_{i,k}$ alors x^{k+1} est sur la droite passant par x^k et perpendiculaire à $H_{i,k}$ l'hyperplan frontière de $C_{i,k}$ défini ainsi $H_{i,k} = \{x \in \mathbb{R}^n ; \langle a_{i,k}, x \rangle - b_{i,k} = 0\}$.

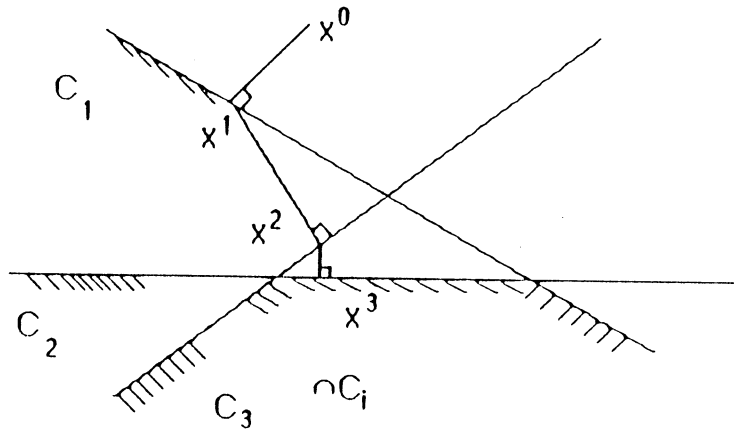


Fig 2. La méthode de Agmon, Motzkin et Schœnberg.

Si le paramètre de relaxation est égal à 1 ($\lambda^k=1$) pour $k \geq 0$, x^{k+1} est la projection orthogonale de x^k sur $C_{i,k}$.

V.10.1.3. METHODE DE RELAXATION DE GUBIN ET AL :

Le problème que traite GUBIN et al[14] est la recherche d'un point commun à une famille de convexes fermés C_i , $i \in I$, où I est ensemble d'indices quelconque.

cette méthode est une généralisation de celle de KACZMARZ[12] (V.10.1.1) et celle de AGMON, MOTZKIN et SHCENBERG[20] (V.10.1.2).

Le processus est très simple et consiste à projeter successivement le point x^k sur un ensemble $C_{i,k}$, $i, k=0, \dots, m$ et prendre un point dans le segment qui joint x^k et son projeté sur $C_{i,k}$.

Anisi, la méthode génère une suite de points de la manière suivante :

- a) Choisir $x^0 \in \mathbb{R}^n$ arbitrairement.
- b) A l'itération k on est en x^k , on détermine i, k par une des deux méthodes suivantes :
 - a.1) La méthode cyclique :
Supposons que $I = \{1, \dots, m\}$ et on prend $i, k = k+1 \pmod{m}$ i.e., i, k est le reste de la division de $k+1$ par m .
 - a.2) La méthode de la distance maximale :
Soit i, k l'indice du convexe le plus éloigné de x^k , i.e.,

$$d(x^k, C_{i,k}) \geq d(x^k, C_i) \quad \forall i \in I.$$

c) On détermine x^{k+1} par :

$$x^{k+1} = x^k + \lambda^k [P_{C_{i,k}}(x^k) - x^k] \quad \text{avec } 0 < \varepsilon_1 \leq \lambda^k \leq 2 - \varepsilon_2 \quad \varepsilon_2 > 0.$$

Retour à b) avec $k \leftarrow k+1$.

Remarque : Si $\lambda^k = 1$ alors x^{k+1} est la projection orthogonale de x^k sur $C_{i,k}$.

THEOREME DE CONVERGENCE 1 : Soient $(C_i)_{i \in I}$ une famille de convexes, $C = \{\cap C_i ; i \in I\} \neq \emptyset$ et $0 < \varepsilon_1 \leq \lambda^k \leq 2 - \varepsilon_2 \quad \varepsilon_2 > 0$.

a) $C_{i^*} \cap ((\cap C_i ; i \in I \text{ et } i \neq i^*))^0$ (où C_{i^*} est un convexe choisi au préalable et A^0 représente l'intérieur de A)

b) Tous les C_i à l'exception de C_{i^*} sont des uniformément convexe.

c) I est fini.

Alors pour un x^0 choisi arbitrairement la suite (x^k) générée par cette méthode converge fortement vers un point $x^* \in \{\cap C_i ; i \in I\}$.

ALGORITHME A CONVERGENCE FINIE : Nous présentons maintenant un algorithme du à GUBIN et al[14], pour la recherche d'un point commun à une famille de convexes fermés, qui présente la particularité de converger en nombre fini d'itérations.

Le processus itératif de cet algorithme est conforme à la procédure suivante:

$$x^{k+1} = P_{C_{i^*}}[x^k + \lambda_{i,k} u_{i,k} / \|u_{i,k}\|^2],$$

où $u_{i,k} = P_{C_{i,k}}(x^k) - x^k$, $i, k \in I(x^k) = \{i \in I ; \|P_{C_{i,k}}(x^k) - x^k\|^2 = \sup(\|P_{C_i}(x^k) - x^k\|^2 ; i \in I)\}$ et i^* est l'indice d'un convexe choisi au préalable.

THEOREME DE CONVERGENCE 2 : Soient $(C_i)_{i \in I}$ une famille de convexes fermés. Si

$$a) C_{i^*} \cap ((\cap C_i ; i = 1, \dots, m \text{ et } i \neq i^*))^0 \neq \emptyset$$

et

$$b) \lambda^k \rightarrow 0 \quad \text{et} \quad \sum_{1 \leq i \leq m} \lambda^k = \infty$$

alors la méthode présentée ci-dessus converge en nombre fini d'itérations, i.e., $\exists k$ tel que $x^k \in (\cap C_i)_{i \in I}$.

V.10.3. METHODE DU TYPE BARYCENTRIQUE :

V.10.3.1. PROCEDE DE CIMMINO :

Le problème est :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } \langle a_i, x \rangle = b_i \quad i=1, 2, \dots, m.$$

où $a_i \in \mathbb{R}^n$ et $b_i \in \mathbb{R}$ pour $i=1, 2, \dots, m$.

Le procédé de CIMMINO[12] consiste à construire une suite $x^0, x^1, \dots, x^k, \dots$ à partir d'un point arbitraire $x^0 \in \mathbb{R}^n$ en deux phases successives:

la première phase consiste à projeter orthogonalement sur les m hyperplans $C_i, C_i = \{x \in \mathbb{R}^n, \langle a_i, x \rangle = b_i\}$, les mêmes points x^k , ce qui donne les points $y_{i,k}$:

$$y_{i,k} = P_{C_i}(x^k) \quad i=1, 2, \dots, m.$$

Avec $P_{C_i}(x^k) = x^k - (b_i - \langle a_i, x^k \rangle) / \|a_i\|^2 a_i$.

Puis à prendre les symétriques de x^k par rapports aux hyperplans, d'où les points :

$$y'_{i,k} = -x^k + 2P_{C_i}(x^k) = x^k + 2(b_i - \langle a_i, x^k \rangle) / \|a_i\|^2 a_i \quad i=1, 2, \dots, m.$$

Quand à la deuxième phase, elle consiste à calculer le barycentre pondéré avec des poids $m_i > 0$ tel que $\sum_{1 \leq i \leq m} \lambda_i = 1$ des points $y'_{i,k}$ ce qui définit le nouveau itéré:

$$x^{k+1} = \sum_{1 \leq i \leq m} \lambda_i y'_{i,k} = \sum_{1 \leq i \leq m} \lambda_i [x^k + 2(b_i - \langle a_i, x^k \rangle) / \|a_i\|^2 a_i]$$

ce qui est équivalent à

$$x^{k+1} = x^k + 2 \sum_{1 \leq i \leq m} \lambda_i [P_{C_i}(x^k) - x^k] \quad \text{avec } \sum_{1 \leq i \leq m} \lambda_i = 1 \text{ et } \lambda_i > 0 \text{ pour tout } i.$$

On a donc successivement une phase 1 de calcul des m projections, puis une phase 2 de coordination. Pour la convergence de ce procédé voir N. GASTINEL[12].

V.10.3.2. METHODE DE BARYCENTRE (AUSLENDER) :

A.AUSLENDER[3] généralise le procédé de G.CIMMINO pour la recherche d'un point commun à une famille finie de convexes fermés.

La méthode du barycentre présentée par A.AUSLENDER consiste à construire à partir d'un point x^0 arbitraire les suites (x^k) et $(x^{i,k})$ $i=1, \dots, m$; $k=0, 1, \dots$ de la façon suivante, supposons calculé x^k alors : on pose

$$x^{i,k} = P_{C_i}(x^k) \quad i=1,2,\dots,m.$$

Puis on calcule l'itéré x^{k+1} par :

$$x^{k+1} = 1/\lambda \sum_{1 \leq i \leq m} x^{i,k} = 1/\lambda \sum_{1 \leq i \leq m} P_{C_i}(x^k)$$

La différence est claire entre cette méthode et le procédé de G.CIMMINO .

THEOREME 1 : La suite (x^k) ainsi définie converge vers un point $x^* \in C$.

Preuve : voir A.AUSLENDER[3] .

G.PIERRA[21] présente la méthode du Barycentre avec poids pondéré qui est conforme à la procédure suivante :

$$x_{i,k} = P_{C_i}(x^k)$$

puis on détermine l'itéré x^{k+1} par .

$$x^{k+1} = \sum_{1 \leq i \leq m} \lambda_i x_{i,k} \quad \text{avec } \sum_{1 \leq i \leq m} \lambda_i = 1 \text{ et } \lambda_i > 0 \quad \forall i.$$

L'extention aux barycentres avec poids de tous les résultats pour les barycentres pondérés est immidiate.

V.10.3.3. METHODE DE YAIR CENSOR :

Y.CENCOR[8] présente une généralisation du procédé de G.CIMMINO au cas des systèmes d'inégalités linéaires.

V.10.3.3.1. POSITION DU PROBLEME ET HYPOTHESES :

Soit le problème :

Trouver $x \in \mathbb{R}^n$ tel que $\langle a_i, x \rangle \leq b_i$ pour $i=1, 2, \dots, m$,

où a_i et x dans \mathbb{R}^n . Pour tout i on définit le demi-espace fermé

$C_i = \{x \in \mathbb{R}^n ; \langle a_i, x \rangle \leq b_i\}$ et l'hyperplan $H_i = \{x \in \mathbb{R}^n ; \langle a_i, x \rangle = b_i\}$.

Soit $C = \bigcap_{1 \leq i \leq m} C_i$

On suppose que $C \neq \emptyset$, i.e., on suppose qu'il existe une solution du problème posé. Pour $x^k \in \mathbb{R}^n$ et $i=1, 2, \dots, m$ on définit :

$$c_{i,k} = \min\{0, (b_i - \langle a_i, x^k \rangle) / \|a_i\|^2\}$$

$$I_k = \{i / c_{i,k} < 0\}.$$

On pose

$$\mu^k = \begin{cases} \sum_{i \in I_k} m_i & \text{si } |I_k| \geq 2 \\ 1 & \text{si } |I_k| = 1 \end{cases}$$

avec $|I_k|$ le cardinal de I_k . Si $I_k = \emptyset$ alors x^k est solution du problème.

DESCRIPTION DE LA METHODE :

Initialisation : $x^0 \in \mathbb{R}^n$ choisi arbitrairement.

Etape principale : On calcule I_k , si $I_k = \emptyset$ alors x^k est une solution .

Sinon on calcule l'itéré x^{k+1} ainsi :

$$x^{k+1} = x^k + 2 / \mu^k (\sum_{1 \leq i \leq m} m_i [P_{C_i}(x^k) - x^k]).$$

La convergence de cet algorithme est démontrée dans Y.CENCOR[8] .

V.10.3.4. METHODE DE PIERRO :

La méthode de DE PIERRO[10] est destinée à résoudre un système d'inégalités, elle est basée sur le principe du procédé de G.CIMMINO. La différence de cette méthode avec celle de Y.CENSOR est que celle de DE PIERRO converge, à partir d'un point arbitraire, vers une solution du système d'inégalités linéaires dans le cas où le système est consistant et vers une solution "aux moindres carrées" dans le cas général.

V.10.3.4.1. POSITION DU PROBLEME :

On considère le système d'inégalités linéaires :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } \langle a_i, x \rangle \leq b_i \text{ pour tout } i=1, 2, \dots, m,$$

où $a_i, x \in \mathbb{R}^n$ ($a_i \neq 0$), $b_i \in \mathbb{R}$ avec $m \geq 2$. Soient $\lambda_1, \dots, \lambda_m$ des nombres réels tels que :

$$\sum_{1 \leq i \leq m} \lambda_i = 1 \text{ avec } \lambda_i > 0 \text{ pour tout } i=1, \dots, m.$$

On définit pour $x^* \in \mathbb{R}^n$ et tout i tel que $1 \leq i \leq m$

$$c_i(x^k) = \min\{0, (b_i - \langle a_i, x^k \rangle) / \|a_i\|^2\}, \text{ alors}$$

$$P(x^k) = x^k + c_i(x^k) a_i$$

définit la projection orthogonale de x sur le demi-espace fermé

$$C_i = \{x \in \mathbb{R}^n ; \langle a_i, x \rangle \leq b_i\}.$$

V.10.3.4.2. ALGORITHME : $x^0 \in \mathbb{R}^n$ choisi arbitrairement.

L'itéré x^{k+1} est donné par :

$$x^{k+1} = x^k + \alpha \sum_{1 \leq i \leq m} \lambda_i [P_{C_i}(x^k) - x^k] \quad \text{avec } 0 < \alpha < 2.$$

Dans le cas général le processus itératif donné par DE PIERRO est :

$$x^{k+1} = x^k + \alpha_k \sum_{1 \leq i \leq m} \lambda_i [P_i(x^k) - x^k] \quad 0 < \varepsilon_1 \leq \alpha_k \leq 2 - \varepsilon_2 \text{ où } \varepsilon_2 > 0.$$

THEOREME 1 [10] : Si $C = \bigcap C_i \neq \emptyset$ alors $\{x^k\}$ définie par l'algorithme converge pour tout $x^0 \in \mathbb{R}^n$ vers un point x^* de C . Si $C = \emptyset$ alors l'algorithme converge vers une solution du système "aux moindres carrés" :

$$f(x^*) = \min \left\{ \sum_{1 \leq i \leq m} \lambda_i \|P_i(x) - x\| ; x \in \mathbb{R}^n \right\}.$$

V.10.3.5. METHODE DE J.E.SPINGARN :

L'algorithme de J.E.SPINGARN[29] pour la résolution d'un système d'inégalités est un cas spécial de la méthode de l'inverse partiel qu'il a introduite. Bien qu'on s'attendait à ce que cette méthode n'ait aucune relation avec les méthodes de projections, on s'est aperçu qu'elle en fait partie. Sa particularité est qu'elle donne en plus de la solution primale des solutions duales. En outre, si l'intérieur relatif de l'intersection de ces demi-espaces fermés est non vide, alors l'algorithme converge en un nombre fini d'itérations vers la solution du problème.

V.10.3.5.1. POSITION DU PROBLEME :

On considère le système d'inégalités linéaires :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } \langle a_i, x \rangle \leq b_i \quad \text{pour tout } i=1, 2, \dots, m, \quad (\mathbf{P}_{10.3.5})$$

où $a_i, x \in \mathbb{R}^n$ ($a_i \neq 0$), $b_i \in \mathbb{R}$ avec $m \geq 2$.

V.10.3.5.2. RAPPEL SUR LA METHODE DE L'INVERSE PARTIEL :

Pour bien situer le problème ($\mathbf{P}_{10.3.5}$) dans le contexte des problèmes de la recherche d'un zéro d'une somme d'opérateurs monotones maximaux, on rappelle quelques propriétés, déjà données au chapitre II, des opérateurs monotones maximaux ainsi que la méthode de l'inverse partiel.

Une multi-application $T: \mathbb{R}^n \mapsto \mathbb{R}^n$ est dite monotone si :

$$\langle x-x', y-y' \rangle \geq 0 \text{ pour } y \in T(x), y' \in T(x').$$

Le graphe de T est l'ensemble $Gr(T) = \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^n ; y \in T(x)\}$. Si le graphe de T n'est contenu dans aucun autre graphe d'un opérateur monotone, T est dit maximal.

Le problème fondamental des opérateurs maximaux monotones est:

$$\text{" Trouver } x \in \mathbb{R}^n \text{ tel que } 0 \in T(x) \text{ " . } \quad (P_{10.1})$$

Soient A, B deux sous-espaces complémentaires orthogonaux de \mathbb{R}^n (i.e., $A^\perp = B$ et $B^\perp = A$). Tout élément $z \in \mathbb{R}^n$ s'écrit d'une manière unique comme $z = z_A + z_B$ avec $z_A \in A$ et $z_B \in B$.

J.P. SPINGARN [29] a introduit la notion d'inverse partiel d'un opérateur T associé à A , noté $T_A: \mathbb{R}^n \mapsto \mathbb{R}^n$ dont le graphe est:

$$Gr(T_A) = \{(x_A + y_B, y_A + x_B) : y \in T(x)\}.$$

C'est à dire $v \in T_A(w)$ si et seulement si $w = x_A + y_B$ et $v = y_A + x_B$.

T_A est monotone (resp maximal) si et seulement si T est monotone (resp maximal). On remarque que :

$$\text{pour tout } x \in A \text{ et } y \in B, 0 \in T_A(x+y) \text{ si et seulement si } y \in T(x).$$

Ainsi $0 \in T_A(z)$ implique $x = z_A$ et $y = z_B$ sont solutions du problème:

$$\text{" Trouver } x \in A \text{ et } y \in B \text{ tel que } y \in T(x) \text{ " } \quad (P_{10.2}) .$$

L'opérateur inverse partiel ainsi défini représente un outil efficace pour la résolution du problème $(P_{10.2})$.

V.10.3.5.3. PRESENTATION DE LA METHODE DE L'INVERSE PARTIEL:

Il s'agit d'appliquer l'algorithme du point approximant à l'opérateur inverse partiel T_A .

Le processus itératif pour résoudre $(P_{10.2})$ se présente comme suit:

étape 1 : $x^0 \in A, y^0 \in B$ choisis arbitrairement.

étape 2 : trouver $x^k \in \mathbb{R}^n, y^k \in \mathbb{R}^n$ tels que : $x^k + y^k = x^k + y^k$ et $y^k \in T(x^k)$.

étape 3 : $x^{k+1} = (x^k)_A$ et $y^{k+1} = (y^k)_B$.

retour à l'étape 2 avec ces nouvelles valeurs .

Le théorème de convergence de cet algorithme (voir [29]) est le suivant :

THEOREME 1 : Soient (x^k) et (y^k) deux suites générées par l'algorithme précédent. On a soit

i) $x^k \rightarrow x^*$ et $y^k \rightarrow y^*$ faiblement, pour x^* et y^* solution de $(P_{10.2})$; soit

ii) $\|x^k + y^k\|^2 \rightarrow \infty$ et $(P_{10.2})$ n'admet pas de solution .

ALGORITHME :

posons $C_i = \{x \in \mathbb{R}^n : \langle a_i, x \rangle \leq b_i\}$ $i=1, \dots, m$.

Soit le problème $(P_{10.3})$ suivant :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } x \in \bigcap_{i=1}^m C_i$$

Si on pose $T_i = \partial \chi_{C_i}$, alors le problème $(P_{10.3})$ s'écrit :

$$\text{Trouver } x \in \mathbb{R}^n \text{ tel que } 0 \in T_1(x) \times T_2(x) \times \dots \times T_m(x), \quad (P_{10.4})$$

pour mettre le problème $(P_{10.4})$ sous la forme $(P_{10.2})$ on définit:

$$\mathcal{X} = \mathbb{R}^n \times \mathbb{R}^n \times \dots \times \mathbb{R}^n, \quad (m \text{ fois})$$

$$T = \partial \chi_{C_1} \times \partial \chi_{C_2} \times \dots \times \partial \chi_{C_m},$$

$$A = \{(x_1, x_2, \dots, x_m) \in \mathcal{X} : x_1 = \dots = x_m\},$$

$$B = \{(y_1, y_2, \dots, y_m) \in \mathcal{X} : y_1 + \dots + y_m = 0\}.$$

On constate que

$(y_1, \dots, y_m) \in T(x_1, \dots, x_m)$ si et seulement si $y_i \in \partial \chi_{C_i}(x_i)$ pour $i=1, \dots, m$.

$T: \mathcal{H} \mapsto \mathcal{H}$ est maximal monotone, A et B sont deux sous-espaces complémentaires orthogonaux de \mathcal{H} ($A=B^\perp$ et $B=A^\perp$).

Dans l'espace produit le problème (2.4) est équivalent à :

$$\begin{aligned} \text{Trouver} \quad & (x_1, \dots, x_m) \in A, \\ & (y_1, \dots, y_m) \in B, \quad (\text{P}_{10.4}) \\ \text{tel que} \quad & (y_1, \dots, y_m) \in T(x_1, \dots, x_m). \end{aligned}$$

Algorithme :

étape 1 : on prend x, y_1, \dots, y_m éléments de \mathbb{R}^n tel que $y_1 + \dots + y_m = 0$.

étape 2 : trouver x_i', y_i' pour $i=1, \dots, m$ éléments de \mathbb{R}^n tels que

$$x_i' = P_{C_i}(x + y_i) \quad \text{et}$$

$$x + y_i = x_i' + y_i' \quad i=1, \dots, m.$$

étape 3 :

$$x^+ = \frac{1}{m} \sum_{i=1}^m P_{C_i}(x + y_i)$$

$$y_i^+ = y_i' - \frac{1}{m} \sum_{i=1}^m y_i' \quad \text{pour } i=1, \dots, m.$$

retour à l'étape 2 avec ces nouvelles valeurs.

Appliquons le théorème (1) dans ce contexte, on obtient le corollaire de convergence de l'algorithme ci-dessus.

COROLLAIRE 1 : Pour les deux suites (x^k) et (y^k) générées par l'algorithme précédent. On a soit

i) $x^k \mapsto x^* \in C$ et $y_i^k \mapsto y_i^* \in \partial \chi_{C_i}$ faiblement avec $y_1^* + \dots + y_m^* = 0$; soit

ii) $\|(x^k + y_1^k, \dots, x^k + y_m^k)\|^2 \mapsto \infty$ et le système d'inégalités n'admet pas de solution.

THEOREME 2 : L'algorithme précédent converge après un nombre fini d'itérations vers une solution du problème $(P_{10.3})$, à condition que $\text{int}C \neq \emptyset$ (intérieur de C est non vide avec $C = \bigcap C_i$)

Preuve : voir J.E.SPINGARN[29].

V.11. CLASSIFICATION DES METHODES DE PROJECTION :

Cette partie est consacrée à la classification des méthodes présentées dans les paragraphes précédents.

Dans le tableau 1, on donne les trois méthodes destinées à la résolution du problème de l'intersection de deux convexes fermés, ou leur distance, lorsque cette intersection est vide.

Dans le tableau 2, on donne les formulations du problème de la recherche d'un point commun à une famille de convexes fermés, en termes d'optimisation, ainsi que leurs méthodes de résolution correspondantes en faisant les liens entre elles.

Dans le tableau 3, on expose les méthodes de projection successives pour la recherche d'un point commun à une famille de convexes fermés, en les rattachant, chaque fois que cela est possible, aux méthodes du type sous-gradient données dans le tableau 2.

Dans le tableau 4, on présente les méthodes barycentriques pour la recherche d'un point commun à une famille de convexes fermés en les liant aux méthodes du type sous-gradient données dans le tableau 2.

Méthodes de projections pour la recherche de l'inter. ou la distance de deux convexes fermés.

Type de méthode	Position du problème	Méthode de résolution	Remarque
Méthode de gradient projeté (1)	$\min\{1/2\ x - P_{C_2}(x)\ ^2, x \in C_1\}$	$x^{k+1} = x^k + \lambda^k [P_{C_1 \cap C_2}(x^k) - x^k]$ $0 < \lambda^k < 2$	Pour le cas un des deux convexes est variété linéaire on a le pas optimal $\lambda^k = 1$
Méthode de projections successives (2)	$\min\{1/2\ x_1 - x_2\ ^2, (x_1, x_2) \in C_1 \times C_2\}$	$x_1^{k+1} = P_{C_1}(x_2^k)$ $x_2^{k+1} = P_{C_2}(x_1^{k+1})$	cas particulier de (1) avec $\lambda^k = 1 \quad \forall k > 0$
Méthode de barycentre (3)	$\min\{\sum_{i=1}^2 \lambda_i \ x_i - y\ ^2; x_i \in C_i, 1 \leq i \leq 2, y \in \mathbb{R}^n\}$	$x_1^{k+1} = P_{C_1}(y^k)$ $x_2^{k+1} = P_{C_2}(y^k)$ $y^{k+1} = \sum_{i=1}^2 \lambda_i P_{C_i}(y^k)$ $\sum_{i=1}^2 \lambda_i = 1$ avec $\lambda_i > 0 \quad i=1,2.$	C'est une méthode nouvelle pour la recherche de l'intersection ou la distance de deux convexes fermés

TABEAU 1

pe de méthodes	Le problème , le processus itératif donné par la méthode associée.	Remarques
Méthode du gradient projeté (1)	$\min\{1/2\ \alpha - P_{\mathcal{E}}(\alpha)\ ^2, \alpha \in \mathcal{D}\}$ $\alpha^{k+1} = P_{\mathcal{D}}[\alpha^k + \mu^k (P_{\mathcal{E}}(\alpha^k) - \alpha^k)]$ $x^{k+1} = x^k + \mu^k \sum_{i=1}^m \lambda_i [P_{C_i}(x^k) - x^k] \quad 0 < \mu^k < 2 \quad \forall k \text{ avec } \sum_{i=1}^m \lambda_i = 1, \lambda_i > 0$	même processus que (3) et (4)
Méthode de décomposition (2)	$\min\{1/2 \sum \lambda_i \ x_i - y\ ^2, x_i \in C_i, i=1, \dots, m, y \in \mathbb{R}^n\}$ $x_i^{k+1} = P_{C_i}(y^k), y^{k+1} = \sum \lambda_i P_{C_i}(y^k) = \sum \lambda_i x_i^{k+1} \text{ avec } \sum \lambda_i = 1, \lambda_i > 0$	cas particulier de (1), (3) et (4) en prenant $\mu^k = 1 \quad \forall k$
Méthode du gradient (3)	$\min\{1/2 \sum \lambda_i \ x - P_{C_i}(x)\ ^2; x \in \mathbb{R}^n\}$ $x^{k+1} = x^k + \mu^k \sum \lambda_i [P_{C_i}(x^k) - x^k] \text{ avec } \sum \lambda_i = 1 \text{ et } \lambda_i > 0 \quad 1 \leq i \leq m.$	même processus que (1) et (4)
Méthode du sous gradient (4)	$\min\{\sup(1/2 d^2(\alpha, \mathcal{E}), 1/2 d^2(\alpha, \mathcal{D}))\}; \alpha \in \mathcal{D}$ $\alpha^{k+1} = \alpha^k + \mu^k [P_{\mathcal{D}} P_{\mathcal{E}}(\alpha^k) - \alpha^k] \quad 0 < \mu^k < 2$ $x^{k+1} = x^k + \mu^k \sum \lambda_i [P_{C_i}(x^k) - x^k] \text{ avec } \sum \lambda_i = 1 \text{ et } \lambda_i > 0 \quad 1 \leq i \leq m.$	même processus que (1) et (3)
Méthode du sous gradient (5)	$\min\{\sup\{1/2 d^2(x, C_i); 1 \leq i \leq m\}; x \in \mathbb{R}^n\}$ $x^{k+1} = x^k + \mu^k [P_{C_{i,k}}(x^k) - x^k]; i, k \in I(x^k) = \{s \in I; d^2(x^k, C_s) = \sup d^2(x^k, C_i)\}$ <p>avec $I = \{1, 2, \dots, m\}$</p>	
Méthode du minimax (6)	$\min \sup\{1/2 d^2(x, C_i); (x, i) \in \mathbb{R}^n \times I\} = \sup \min\{1/2 d^2(x, C_i); (i, x) \in I \times \mathbb{R}^n\}$ $x^{k+1} = P_{C_{i,k}}(x^k) \quad i, k \in I(x^k) = \{s \in I; d^2(x^k, C_s) = \sup\{d^2(x^k, C_i); i \in I\}\}$ <p>avec $I = \{1, 2, \dots, m\}$</p>	cas particulier de (5) en prenant $\mu^k = 1 \quad \forall k > 0$.

TABLEAU 2

Méthode de projections successives.

Type de méthode	Problème	Processus itératif donné par la méthode associée	Remarque
Méthode de Kaczmarz	Trouver $x \in \mathbb{R}^n$ tel que $\langle a_i, x \rangle = b_i$ pour $1 \leq i \leq m$	$x^{k+1} = x^k + \lambda \frac{b_{i,k} - \langle a_{i,k}, x^k \rangle}{\ a_{i,k}\ ^2} a_{i,k}$ $0 < \lambda^k < 2 \quad \forall k; i, k = k+1 [m]$	Résolution d'un système d'équations linéaires.
Méthode de Agmon	Trouver $x \in \mathbb{R}^n$ tel que $\langle a_i, x \rangle \leq b_i$ pour $1 \leq i \leq m$	$x^{k+1} = x^k + c^k a_{i,k}$ $c^k = \min\left\{0, \lambda^k \frac{b_{i,k} - \langle a_{i,k}, x^k \rangle}{\ a_{i,k}\ ^2}\right\}$ $i, k = k+1 [m]$	Résolution d'un système d'inégalités linéaires.
Méthode de Gubin et al	Trouver $x \in \mathbb{R}^n$ tel que $x \in \bigcap_{i=1}^m C_i$ où les C_i sont des convexes fermés pour $i=1, 2, \dots, m$.	$1) x^{k+1} = x^k + \lambda^k [P_{C_{i,k}}(x^k) - x^k]$ $0 < \lambda^k < 2 \quad i, k = k+1 [m]$ $2) x^{k+1} = x^k + \lambda^k [P_{C_{i,k}}(x^k) - x^k]$ $0 < \lambda^k < 2$ $i, k \in I(x^k) = \{i / d(x^k, C_i) \geq d(x^k, C_j), 1 \leq j \leq m\}$	recherche d'un point commun à une famille de convexes fermés quelconques. même processus que celui de (5) et (6) de la figure 2 de la page précédente.

TABEAU 3

Méthodes barycentriques.

Type de méthode	Position du problème	Méthode de résolution	Remarque
Méthode de Cimmino	trouver $x \in \mathbb{R}^n$ tel que $\langle a_i, x \rangle = b_i \quad 1 \leq i \leq m.$	$x^{k+1} = x^k + 2 \sum_{i=1}^m m_i [P_{C_i}(x^k) - x^k]$ $\sum m_i = 1$ avec $m_i > 0$ pour $1 \leq i \leq m.$	Cas particulier du processus (1), (3) et (4) du tab. 2 $\mu^k = 1$
Méthode de barycentre	trouver $x \in \mathbb{R}^n$ tel que $x \in \bigcap_{i=1}^m C_i$ avec C_i sont des convexes fermés pour $1 \leq i \leq m.$	$x^{k+1} = \frac{1}{m} \sum_{i=1}^m P_{C_i}(x^k)$	Cas particulier de (1), (3) et (4) du tab. 2 en prenant $\lambda_i = 1 \forall i$ et $\mu^k = 1 \forall k$
Méthode de Y. Censor	Trouver $x \in \mathbb{R}^n$ tel que $\langle a_i, x \rangle \leq b_i \quad 1 \leq i \leq m$	$x^{k+1} = x^k + \frac{2}{\mu^k} \left[\sum_{i=1}^m m_i [P_{C_i}(x^k) - x^k] \right]$ $\sum_{i=1}^m m_i = 1$ avec $m_i > 0 \quad 1 \leq i \leq m.$	Cas particulier du processus (1), (3) et (4) en prenant $\mu^k = 2/\mu^k$.
Méthode de De Pierro	Trouver $x \in \mathbb{R}^n$ tel que $\langle a_i, x \rangle \leq b_i \quad 1 \leq i \leq m$	$x^{k+1} = x^k + \alpha \sum_{i=1}^m \lambda_i [P_{C_i}(x^k) - x^k]$ avec $0 < \alpha < 2$	Cas particulier du processus (1), (3) et (4) avec $\mu^k = \alpha \forall k$

TABLEAU 4

REFERENCES

- [1] **S.AGMON (1969)**
The Relaxation Method for Linear Inequalities.
Can. J. Math. 6, 3, 392-393.
- [2] **A.AUSLENDER (1974)**
Méthodes de décomposition pour la minimisation d'une fonction sur un espace produit. *S.I.A.M J. Control* , 12 , 4 , 635-642.
- [3] **A.AUSLENDER (1976)**
Optimisation méthodes numériques.
Masson, Paris, 1976.
- [4] **L.M.BERGMAN (1965)**
The method of successive projection for finding a common point of convex sets. *Dokl. Akad. Nauk. SSSR*, 162, 3, (1965), 487-490
- [5] **J.CEA (1968)**
Les méthodes de descente en théorie de l'optimisation.
RIRO, 13-R3, (1968), 79-101
- [6] **J.CEA (1971)**
Optimisation.
Dunod, Paris.
- [7] **Y.CENSOR (1981)**
Row-Action Methods for Huge and Sparse Systems and their Applications
S.I.A.M Review Vol 23, N° 4, October 1982.
- [8] **Y.CENSOR and T.ELGIVINGI (1985)**
New Methods for Linear Inequalities.
Linear Algebra and its Applications 42-199-21.
- [9] **V.F.DEM'YANOV and L.V.VASIL'EV(1985)**
Nondifferentiable Optimization.
Optimization Software, Inc. New York.
- [10] **A.R.DE PIERRO and A.N.IUSEM (1985)**
A Simultaneous Projections Methods for Linear Inequalities. *Linear Algebra and its Applications* 64 : 243-253.
- [11] **I.EKELAND et TEMAM(1984)**
Analyse convexe et problèmes variationnels.
Dunod , Paris.

[12] **N.GASTINEL (1966)**

Analyse numérique linéaire.
Hermann, Paris.

[13] **J.L.GOFFIN (1980)**

Nondifferentiable optimisation and the relaxation Method.

[14] **L.G.GUBIN, B.T.POLYAK and E.V.RAIK (1967)**

The Method of Projections for Finding the Common Point of Convex Sets.
Zh. Vychist. Mat. Fiz. , 7, 6, (1967), 1211-1228.

[15] **Y.HAUGAZEAU (1968)**

Sur les inéquations variationnelles et la minimisation de fonctionnelles convexes. Thèse, Paris, (1968).

[16] **P.J.LAURENT (1972)**

Approximation et optimisation.
Editions Hermann, Paris.

[17] **B.MARTINET (1972)**

Algorithmes pour la résolution de problèmes d'optimisation et de minimax.
Thèse, Grenoble, 1972.

[18] **G.MEYER and E.POLAK (1970)**

A Decomposition Algorithm for Solving a Class of Optimal Control Problems.
J. Math. Anal. Appl. , 32, (1970), 118-140.

[19] **M.MICHALOPOULOS (1982)**

Méthodes de sous-gradient dans les problèmes d'optimisation avec contraintes : Application de la programmation linéaire. Thèse, Grenoble.

[20] **T.S.MOTZKIN and L.SHOENBERG (1954)**

The Relaxation Method for Linear Inequalities.
Can. J. Math., 6, 3, 393-404

[21] **G.PIERRA (1976)**

Méthode de décomposition et croisement d'algorithmes pour des problèmes d'optimisation. Thèse, Grenoble, 1976.

[22] **G.PIERRA (1973)**

Sur le croisement de méthode de descente et l'intersection ou la distance de deux convexes. *C. R. Acad. Sci. Paris*, 277, (1973), 1071-1074.

[23] **E.POLAK (1971)**

Computational Methods in Optimization.
Academic Press, New York.

[24] B.T.POLYAK (1967)

*A General Method of Solving Extremum Problems.
Soviet Mathematics Doklady 8 : 593-597.*

[25] B.T.POLYAK (1971)

*Existence Theorems and Convergence of Minimizing Sequences in
Extremum Problems . Dokl. Akad. Nauk. SSSR, 166, 2, 287-290.*

[26] B.POLAJK (1976)

Subgradient Methods : A survey of Soviet Research.

[27] J.P.ROSEN (1960)

*The Gradient Projection Method for non Linear Programming.
SIAM Control, 8, 1, 181-217*

[28] K.H.SLAOUI (1986)

*Application des techniques mathématiques à la gestion des mélanges :
histosplines et optimisation. Thèse, Grenoble, 1986.*

[29] J.E.SPINGARN (1985)

*A primal-Dual Projection Method for solving Systems of Linear Inequalities.
Linear Algebra and its Applications 65 : 54-62.*

CHAPITRE VI

RESULTAS NUMERIQUES ET CONCLUSION.



VI.1. EXPERIMENTATIONS NUMERIQUES :

Les méthodes étudiées dans les chapitres I, II, III et IV ont été programmées sur micral 9050 et sur multics en langage pascal. A partir d'une série d'exemples nous présentons des résultats numériques expérimentaux pour étudier le comportement et la performance d'un tel ou tel algorithme.

On expose deux exemples du problème de régression isotone, et deux autres du problème de régression concave traités dans l'article de C.F.WU.

Les résultats sont regroupés dans des tableaux où figurent les noms des méthodes utilisées, les données initiales ainsi que la solution du problème.

Dans le tableau 1, on présentera la solution du problème de régression isotone de taille 12 calculée par la méthode de KRUSKAL, la méthode de pivotage complémentaire, la méthode du gradient conjugué et la méthode de DYKSTRA.

Dans le tableau 2, on présentera les résultats, en appliquant les méthodes citées ci-dessus, au problème de régression isotone mais pour une dimension égale à 20. Il faut remarquer que la méthode de DYKSTRA pour ce cas boucle.

Dans le tableau 3, on donne les résultats numériques, en traitant un exemple de régression concave de dimension 5, par les méthodes de pivotage complémentaire, du gradient conjugué, de DYKSTRA et de l'inverse partiel.

Dans le tableau 4, les résultats numériques d'un exemple de régression concave de dimension 8, sont présentés par application des méthodes de pivotage complémentaire, du gradient conjugué, de DYKSTRA et de l'inverse partiel. On voit la faiblesse de la méthode de DYKSTRA même dans un cas de petite taille.

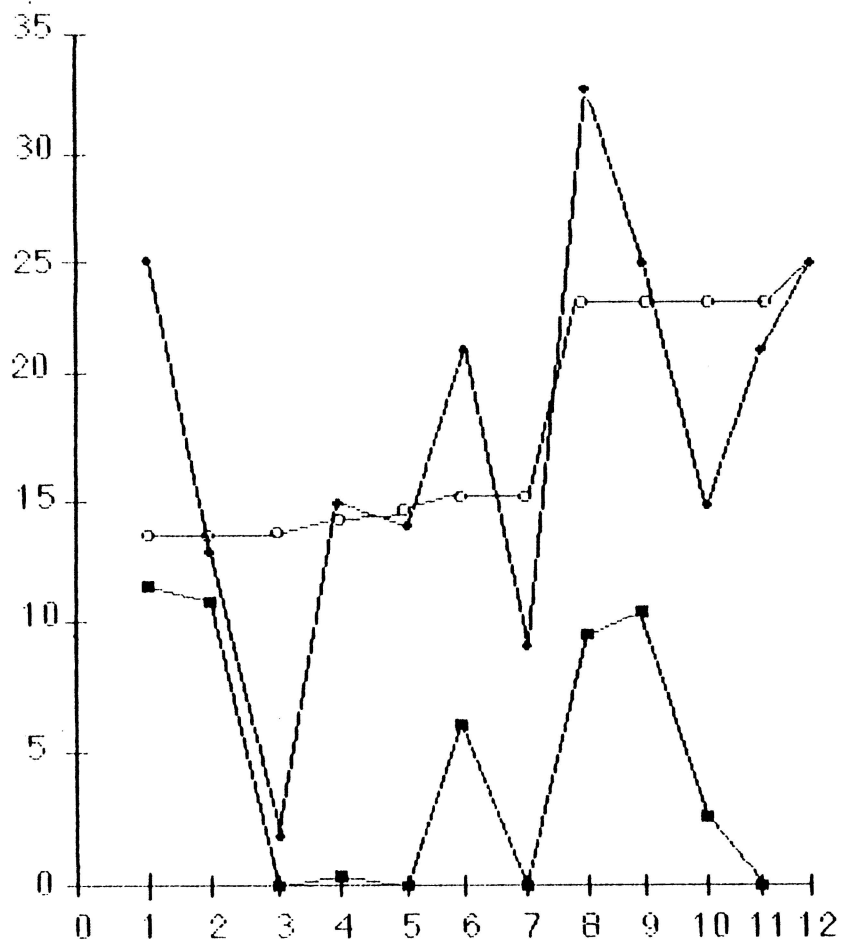
Dans chaque cas les résultats sont illustrés sur des figures.

TABLE 1

poids	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
données	25	13	2	15	14	21	9	33	25	15	21	25
PAV solution	13.3	13.3	13.3	14.5	14.5	15	15	23.5	23.5	23.5	23.5	25
Grad.conj	13.3	13.3	13.3	14.5	14.5	15	15	23.5	23.5	23.5	23.5	25
LEMKE	λ^*	11.67	11.33	0.0	0.5	0.0	6	0	9.5	11	2.5	0
	x^*	13.3	13.3	13.3	14.5	14.5	15	15	23.5	23.5	23.5	23.5
DYKSTRA	13.3	13.3	13.3	14.5	14.5	15	15	23.5	23.5	23.5	23.5	25

x^* désigne la solution primale du problème de régression isotone.

λ^* désigne la solution du problème dual associé.



- ● - données initiales
 - ○ - PAV solution. LEMKE
 DYKSTRA. Grad.conj.
 - ■ - solu. duale donnée par
 l'algo. de LEMKE.

FIGURE 1

TABLE 2

sol.prim = solution primale, sol.duale = solution duale.

weights données		PAV alg	Grad.conj	LEMKE	Inverse Partiel		
		sol. prim	sol.prim	[sol. duale sol. prim]	[sol. duale	sol. prim]	
1	25	13.3	13.3	11.67	13.3	11.70	13.30
1	13	13.3	13.3	11.33	13.3	11.40	13.30
1	2	13.3	13.3	0.00	13.3	0.10	13.30
1	15	14.5	14.5	0.50	14.5	0.40	14.50
1	14	14.5	14.5	0.00	14.5	0.00	14.48
1	21	15.0	15.0	6.00	15.0	6.00	15.00
1	9	15.0	15.0	0.00	15.0	0.00	15.00
1	33	19.3	19.3	13.69	19.3	13.68	19.30
1	25	19.3	19.3	19.38	19.3	19.40	19.28
1	15	19.3	19.3	15.08	19.3	15.08	19.32
1	21	19.3	19.3	16.77	19.3	16.77	19.30
1	25	19.3	19.3	22.46	19.3	22.46	19.30
1	19	19.3	19.3	22.15	19.3	22.15	19.30
1	17	19.3	19.3	19.85	19.3	19.85	19.30
1	9	19.3	19.3	9.54	19.3	9.54	19.30
1	31	19.3	19.3	21.33	19.3	21.33	19.30
1	26	19.3	19.3	27.92	19.3	27.92	19.30
1	7	19.3	19.3	15.62	19.3	15.62	19.30
1	6	19.3	19.3	2.31	19.3	2.31	19.29
1	17	19.3	19.3		19.3		19.31

- □ - données initiales
- • - solu.prim. LEMKE. PAY.
Inverse partiel. Grad conj.
- ○ - solu.duale LEMKE.
Inverse partiel.

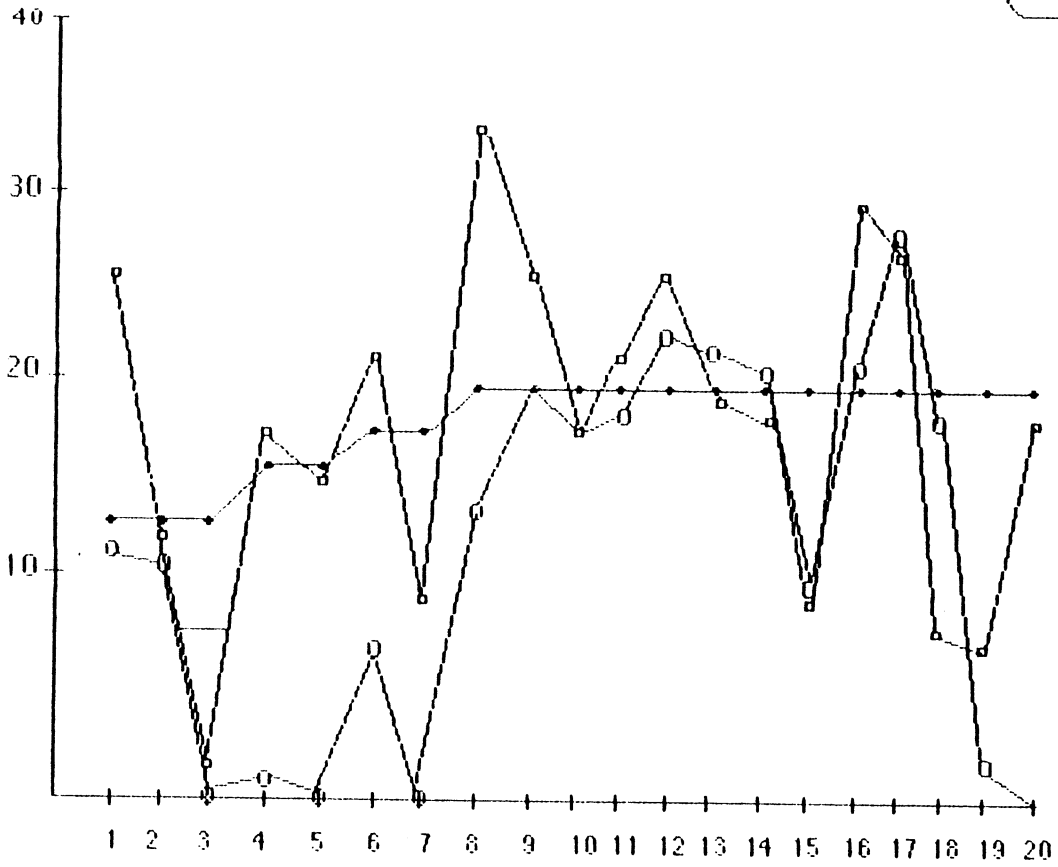


FIGURE 2

TABLE 3

Approximate solutions

weights	abscissa	observations	WU	DYKSTRA	Conj.Grad	LEMKE	
			solu.x*	solu.x*	solu. x*	solu.λ*	solu.x*
1	2	-10	-10.00	-10.000	-10.000	0.00	-10.00
1	4	-2	-3.26	-3.911	-3.260	0.42	-3.26
1	6	-6	-3.90	-3.911	-3.895	0.00	-3.90
1	9	-4	-4.84	-3.911	-4.842		-4.84
1	10	-8	-8.00	-8.000	-8.000		-8.00
Number of iterations					22	4	2
Minimum obtained			6.700	8.023	6.703		6.700

solu.x* désigne la solution primale du problème.

solu.λ* désigne la solution du problème dual associé.

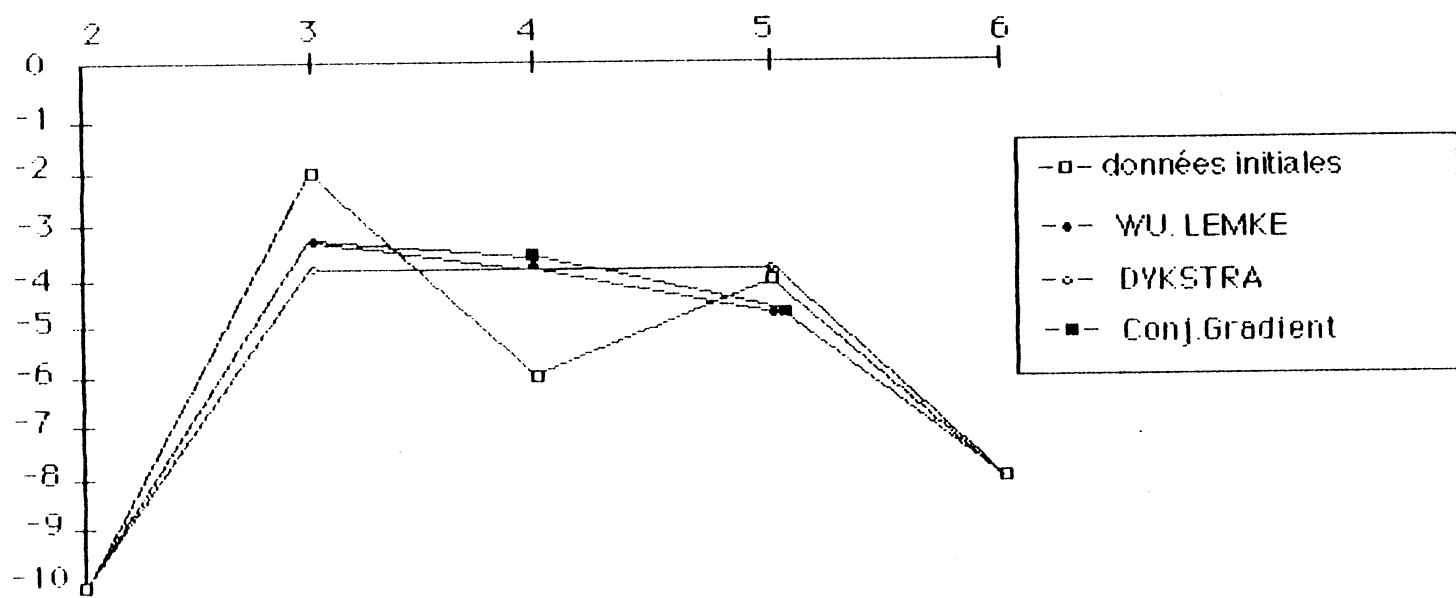


FIGURE 3

TABLE 4

Approximate solutions

weights		abscissa	observations	WU	DYKSTRA	Conj.Grad	LEMKE	
w	y	g	x*	x*	x*	λ^*	x*	
27	00	22.94	22.94	40.07	22.940	0	22.94	
9	20	41.58	41.58	68.63	41.580	0.53	41.58	
8	40	65.48	59.65	68.63	58.293	2.45	59.65	
10	60	58.81	68.11	68.63	69.269	2.17	68.11	
9	80	81.74	74.24	68.63	79.616	2.44	74.24	
19	120	82.15	84.36	68.63	80.356	1.16	84.36	
10	160	96.59	94.49	68.63	95.164		94.49	
8	180	94.04	94.01	68.63	95.905		94.01	
Number of iterations					211	10		5
Minimum obtained				548.69	335.10	549.2		548.69

w désigne la fonction poids, y est la variable telle que $0 < y_1 < \dots < y_8$, g désigne le vecteur des données, x^* désigne la solution primale du problème de régression concave et λ^* désigne la solution du problème dual associé.

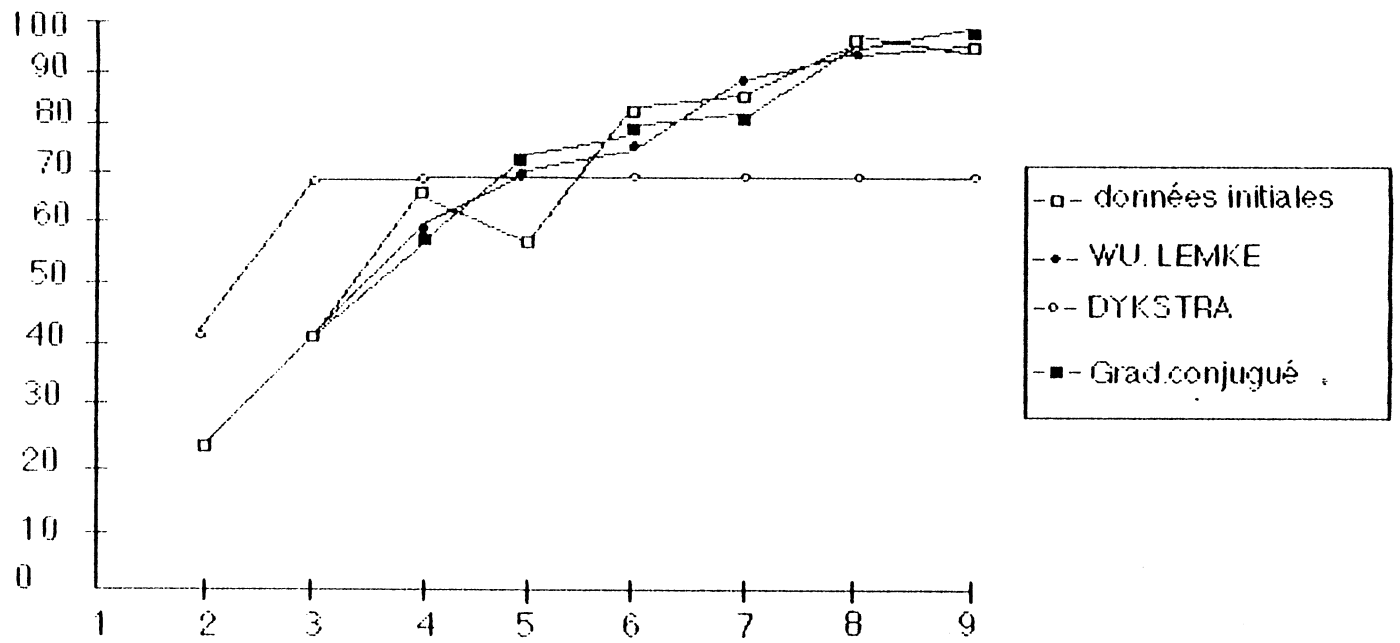


FIGURE 4



CONCLUSION GENERALE



Le travail de cette thèse consiste en un premier temps à :

- étudier des méthodes existantes en littérature.
- proposer de nouveaux algorithmes.
- comparer toutes ces méthodes, et tout cela, pour la résolution de la classe des problèmes de régression.

et en un deuxième temps à :

- donner de nouvelles méthodes basées sur la notion de projections.
- faire un survol des méthodes de projection existantes.
- classer les méthodes de projections et faire les liens entre elles.

Dans la première partie, nous avons formulé le problème de régression en termes d'optimisation, sous forme de trois types de problèmes :

- problème de la recherche d'un zéro d'une somme d'opérateurs monotones maximaux.
- problème de complémentarité linéaire.
- problème de programmation quadratique sous des contraintes linéaires.

Chacune de ces formulations nous a permis d'adapter la méthode d'optimisation convenable à la résolution de chaque problème énuméré ci-dessus, à savoir :

- la méthode de l'inverse partiel.
- la méthode de pivotage complémentaire.
- la méthode du gradient conjugué.

Nous avons étudié ces trois méthodes dans un cadre théorique général et pour pouvoir appliquer notre étude aux problèmes de l'analyse des données, nous avons mis les problèmes de régression sous l'une des formes citées ci-dessus. Nous avons programmé les méthodes les plus utilisées en analyse des données, à savoir, la méthode de Kruskal et la méthode de DYKSTRA ainsi que ces trois méthodes sur micral 9050 et sur multics, en langage pascal.

Nous dirons que les méthodes d'optimisation (la méthode de l'inverse partiel, la méthode de pivotage complémentaire, la méthode du gradient conjugué) sont plus intéressantes, au moins pour la majorité des cas, que celles proposées par les statisticiens et elles permettent de résoudre, relativement à un ordre quelconque, des problèmes de régression, qui demeurent incomplètement résolus.

Les expériences que nous avons présentées nous montrent la nette supériorité des méthodes d'optimisation sur celles données par les statisticiens.

Il convient de signaler que parmi les méthodes d'optimisation proposées, on a constaté que la méthode de pivotage complémentaire l'emporte, du point de vue du coût, sur la méthode du gradient conjugué et surtout, sur la méthode de l'inverse partiel. Cela peut s'expliquer par le fait que la méthode de pivotage est une adaptation de la méthode simpliciale.

Cela ne veut pas dire que l'on nie la performance du gradient conjugué et ses qualités en matière de convergence.

L'avantage de la méthode de l'inverse partiel est qu'elle permette de résoudre une grande classe de problèmes en analyse numérique, et elle est d'autant plus simple à mettre en œuvre que son application proximale est facile à déterminer, ce qui est le cas pour les problèmes de régression.

Dans la deuxième partie, au chapitre V, on a mis le point sur les méthodes de projections, que ce soit pour la recherche de la distance de deux convexes fermés ou pour la recherche de l'intersection d'une famille de convexes fermés. En formulant ces problèmes dans l'espace produit, nous avons donné des méthodes nouvelles qui englobent au moins la majorité des méthodes connues et plus particulièrement celles du type barycentriques.

On reproche toujours aux méthodes de projections de converger lentement et cela s'explique par le fait qu'elles se déduisent des méthodes du sous gradient qui ont cet inconvénient.

On pense que cette étude synthétique et classificatrice sur les méthodes de projection, présente un document de référence dans ce domaine.

AUTORISATION DE SOUTENANCE

DOCTORAT 3ème CYCLE, DOCTORAT-INGENIEUR, DOCTORAT USTMG

Vu les dispositions de l'Arrêté du 16 avril 1974,

Vu les dispositions de l'Arrêté du 5 juillet 1984,

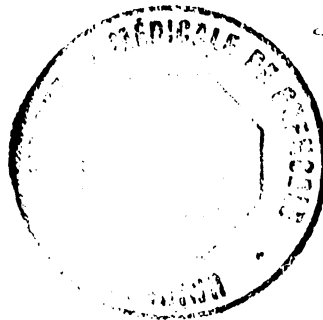
Vu les rapports de M. PHAM DINH TAO

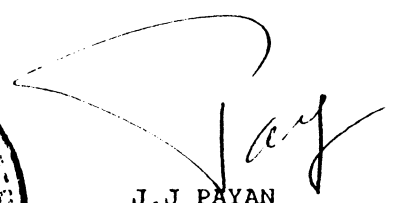
M. DROUET. D'AUBIGNY. Gérard

M. BOUGHAZI Mohamed Ali est autorisé
à présenter une thèse en vue de l'obtention du grade de Docteur
troisième Cycle en Mathématiques Appliquées.
.....

Grenoble, le... 1987

Le Président de l'Université Scientifique
Technologique et Médicale




J. J. PRYAN

