



Une extension du problème du voyageur de commerce : le problème de ramassage scolaire

Kuon Ea

► To cite this version:

Kuon Ea. Une extension du problème du voyageur de commerce : le problème de ramassage scolaire. Modélisation et simulation. Institut National Polytechnique de Grenoble - INPG; Université Joseph-Fourier - Grenoble I, 1978. Français. NNT : . tel-00288677

HAL Id: tel-00288677

<https://theses.hal.science/tel-00288677>

Submitted on 18 Jun 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

Université Scientifique et Médicale de Grenoble

pour obtenir le grade de

DOCTEUR DE 3ème CYCLE
en Mathématiques Appliquées
option : Recherche Opérationnelle

par

Kuon EA



**UNE EXTENSION DU PROBLEME
DU VOYAGEUR DE COMMERCE :
LE PROBLEME DE RAMASSAGE SCOLAIRE.**



Thèse soutenue le 6 décembre 1978 devant la Commission d'Examen :

Président : M. SAKAROVITCH

Examineurs : C. BENZAKEN

J. FONLUPT

P. J. LAURENT

Monsieur Gabriel CAU : Président
Monsieur Joseph KLEIN : Vice-Président

MEMBRES DU CORPS ENSEIGNANT de l'U.S.M.G.

PROFESSEURS TITULAIRES.

MM	AMBLARD Pierre	Clinique de dermatologie
	ARNAUD Paul	Chimie
	ARVIEU Robert	I.S.N.
	AUBERT Guy	Physique
	AYANT Yves	Physique approfondie
Mme	BARBIER Marie-Jeanne	Electrochimie
MM.	BARBIER Jean-Claude	Physique Experimentale
	BARBIER Reynold	Géologie appliquée
	BARJON Robert	Physique nucléaire
	BARNOUD Fernand	Biosynthèse de la cellulose
	BARRA Jean-René	Statistiques
	BARRIE Joseph	Clinique chirurgicale A
	BEAUDOING André	Clinique de Pédiatrie et Puériculture
	BELOREZKY Elie	Physique
	BERNARD Alain	Mathématiques Pures
Mme	BERTRANDIAS Françoise	Mathématiques Pures
MM.	BERTRANDIAS Jean-Paul	Mathématiques Pures
	BEZES Henri	Clinique chirurgicale et Traumatologie
	BLAMBERT Maurice	Mathématiques Pures
	BOLLIET Louis	Informatique (III B)
	BONNET Jean-Louis	Clinique Ophtalmologique
	BONNET-EYMAED Joseph	Clinique Hépatogastro-entérologique
Mme	BONNET Marie-Jeanne	Chimie générale
MM.	BOUCHERLE André	Chimie et Toxicologie
	BOUCHEZ Robert	Physique nucléaire
	BOUSSARD Jean-Claude	Mathématiques appliquées
	BOUTET DE MONVEL Louis	Mathématiques Pures
	BRAVARD Yves	Géographie
	CABANEL Guy	Clinique rhumatologique et hydrologique
	CALAS François	Anatomie
	CARRIER Georges	Biologie végétale
	CARRAZ Gilbert	Biologie animale et pharmacodynamie
	CAU Gabriel	Médecine légale et toxicologie
	CAUQUIS Georges	Chimie organique
	CHABAUTY Claude	Mathématiques Pures
	CHARACHON Robert	Clinique Oto-rhino-laryngologique
	CHATEAU Robert	Clinique de neurologie
	CHIBON Pierre	Biologie animale
	COEUR André	Pharmacie chimique et chimie analytique
	COUDERC Pierre	Anatomie pathologique
	DEBELMAS Jacques	Géologie générale
	DEGRANGE Charles	Zoologie
	DELORMAS Pierre	Pneumophtisiologie

MM.	DEPORTES Charles	Chimie minérale
	DESRÉ Pierre	Métallurgie
	DODU Jacques	Mécanique appliquée (IUT I)
	DOLIQUE Jean-Michel	Physique des plasmas
	DREYFUS Bernard	Thermodynamique
	DUCROS Pierre	Cristallographie
	FONTAINE Jean-Marc	Maths pures.
	GAGNAIRE Didier	Chimie Physique
	GALVANI Octave	Mathématiques pures
	GASTINEL Noël	Analyse numérique
	GAVEND Michel	Pharmacologie
	GEINDRE Michel	Electroradiologie
	GERBER Robert	Mathématiques pures
	GERMAIN Jean-Pierre	Mécanique
	GIRAUD Pierre	Géologie
	JANIN Bernard	Géographie
	KAHANE André	Physique générale
	KLEIN Joseph	Mathématiques pures
	KOSZUL Jean-Louis	Mathématiques pures
	KRAVCHENKO Julien	Mécanique
	LACAZE Albert	Thermodynamique
	LACHARME Jean	Biologie végétale
Mme	LAJZEROWICZ Janine	Physique
MM.	LAJZEROWICZ Joseph	Physique
	LATREILLE René	Chirurgie générale
	LATURAZE Jean	Biochimie Pharmaceutique
	LAURENT Pierre	Mathématiques appliquées
	LEDRU Jean	Clinique médicale B
	LE ROY Philippe	Mécanique (IUT I)
	LLIBOUTRY Louis	Géophysique
	LOISEAUX Jean-Marie	Sciences nucléaires
	LONGUEUE Jean-Pierre	Physique nucléaire
	LOUP Jean	Géographie
Mlle	LUTZ Elisabeth	Mathématiques pures
MM.	MALINAS Yves	Clinique obstétricale
	MARTIN-NÔEL Pierre	Clinique cardiologique
	MAYNARD Roger	Physique du solide
	MAZARE Yves	Clinique Médicale A
	MICHEL Robert	Minéralogie et Pétrographie
	MICOUD Max	Clinique Maladies infectieuses
	MOURIQUAND Claude	Histologie
	MOUSSA André	Chimie nucléaire
	NEGRE Robert	Mécanique
	NOZIERES Philippe	Spectrométrie Physique
	OZENDA Paul	Botanique
	PAYAN Jean-Jacques	Mathématiques pures
	PEBAY-PEYROULA Jean-Claude	Physique
	PERRET Jean	Sémiologie Médicale (Neurologie)
	RASSAT André	Chimie systématique
	RENARD Michel	Thermodynamique
	REVOL Michel	Urologie
	RINALDI Renaud	Physique
	DE ROUGEMONT Jacques	Neuro-Chirurgie
	SARRAZIN Roger	Clinique chirurgicale B
	SEIGNEURIN Raymond	Microbiologie et Hygiène
	SENGEL Philippe	Zoologie
	SIBILLE Robert	Construction mécanique (IUT I)
	SOUTIF Michel	Physique générale
	TANCHE Maurice	Physiologie

MM.	VAILLANT François	Zoologie
	VALENTIN Jacques	Physique Nucléaire
Mme	VERAIN Alice	Pharmacie galénique
MM.	VERAIN André	Physique - Biophysique
	VEYKET Paul	Géographie
	VIGNAIS Pierre	Biochimie médicale

PROFESSEURS ASSOCIES

MM.	CRABBE Pierre	CERMO
	SUNIER Jules	Physique

PROFESSEURS SANS CHAIRE

Mle	AGNIUS-DELOD Claudine	Physique pharmaceutique
	ALARY Josette	Chimie analytique
MM.	AMBROISE-THOMAS Pierre	Parasitologie
	ARMAND Gilbert	Géographie
	BENZAKEN Claude	Mathématiques appliquées
	BIAREZ Jean-Pierre	Mécanique
	BILLET Jean	Géographie
	BOUCHET Yves	Anatomie
	BRUCEL Lucien	Energétique (IUT I)
	BUISSON René	Physique (IUT I)
	BUTEL Jean	Orthopédie
	COHEN-ADDAD Jean-Pierre	Spectrométrie physique
	COLOMB Maurice	Biochimie médicale
	CONTE René	Physique (IUT I)
	DELOBEL Claude	M.I.A.G.
	DEPASSEL Roger	Mécanique des fluides
	GAUTRON René	Chimie
	GIDON Paul	Géologie et Minéralogie
	GLENAT René	Chimie organique
	GROULADE Joseph	Biochimie médicale
	HACQUES Gérard	Calcul numérique
	HOLLARD Daniel	Hématologie
	HUGONOT Robert	Hygiène et Médecine préventive
	IDELMAN Simon	Physiologie animale
	JOLY Jean-René	Mathématiques Pures
	JULLIEN Pierre	Mathématiques Appliquées
Mme	KAHANE Josette	Physique
MM.	KRAKOWIACK Sacha	Mathématiques Appliquées
	KUHN Gérard	Physique (IUT I)
	LUU DUC Cuong	Chimie organique - Pharmacie
	MICHOULIER Jean	Physique (IUT I)
Mme	MINIER Colette	Physique (IUT I)
MM.	PELMONT Jean	Biochimie
	FERRIAUX Jean-Jacques	Géologie et Minéralogie
	PFISTER Jean-Claude	Physique du solide
Mle	PIERY Yvette	Physiologie animale

MM.	RAYNAUD Hervé	M.I.A.G.
	REBECCO Jacques	Biologie (CUS)
	REYMOND Jean-Charles	Chirurgie générale
	RICHARD Lucien	Biologie végétale
Mme	RINAUDO Marguerite	Chimie macromoléculaire
MM.	SARROT-REYNAUD Jean	Géologie
	SIROT Louis	Chirurgie générale
Mme	SOUTIF Jeanne	Physique générale
MM.	STIEGLITZ Paul	Anesthésiologie
	VIALON Pierre	Géologie
	VAN CUTSEM Bernard	Mathématiques appliquées

MAITRES DE CONFERENCES ET MAITRES DE CONFERENCES AGREGES

MM.	ARMAND Yves	Chimie (IUT I)
	BACHELOT Yvan	Endocrinologie
	BARGE Michel	Neuro-chirurgie
	BEGUIN Claude	Chimie organique
Mme	BERIEL Hélène	Pharmacodynamie
MM.	BOST Michel	Pédiatrie
	BOUCHARLAT Jacques	Psychiatrie adultes
Mme	BOUCHE Liane	Mathématiques (CUS)
MM.	BRODEAU François	Mathématiques (IUT B) (Personne étrangère habilitée à être di- recteur de thèse).
	BERNARD Pierre	Gynécologie
	CHAMBAZ Edmond	Biochimie médicale
	CHAMPETIER Jean	Anatomie et organogénèse
	CHARDON Michel	Géographie
	CHERADAME Hervé	Chimie papetière
	CHIAVERINA Jean	Biologie appliquée (EFP)
	COLIN DE VERDIERE Yves	Maths pures
	CONTAMIN Charles	Chirurgie thoracique et cardio-vasculaire
	CORDONNIER Daniel	Néphrologie
	COULOMB Max	Radiologie
	CROUZET Guy	Radiologie
	CYROT Michel	Physique du solide
	DENIS Bernard	Cardiologie
	DOUCE Roland	Physiologie végétale
	DUSSAUD René	Mathématiques (CUS)
Mme	ETERRADOSSI Jacqueline	Physiologie
MM.	FAURE Jacques	Médecine légale
	FAURE Gilbert	Urologie
	GAUTIER Robert	Chirurgie générale
	GIDON Maurice	Géologie
	GROS Yves	Physique (IUT I)
	GUIGNIER Michel	Thérapeutique
	GUILLON Jacques	Chimie
	HICTER Pierre	Chimie
	JALBERT Pierre	Histologie
	JUNIEN-LAVILLAVROY Claude	O.R.L.
	KOLODIE Lucien	Hématologie
	LE NOC Pierre	Bactériologie-virologie
	MACHE Régis	Physiologie végétale
	MAGNIN Robert	Hygiène et médecine préventive
	MALLION Jean-Michel	Médecine du travail

MM. MARECHAL Jean
MARTIN-BOUYER Michel
MASSOT Christian
NEMOZ Alain
NOUGARET Marcel
PARAMELIE Bernard
PECCOUD François

PEFFEN René
PERPIER Guy
PHILIP Xavier
RACHAIL Michel
RACINET Claude
RAMBAUD Pierre
RAPHAEL Bernard

Mme RENAUDET Jacqueline
ROBERT Jean-Bernard
ROMIER Guy

SAKAROVITCH Michel
SCHAEFER René
SEIGLE-MURANDI Françoise
STOEBNER Pierre
STUTZ Pierre
VROUSOS Constantin

MAITRES DE CONFERENCES ASSOCIES.

MM. DEVLNE Roderick
KANEKO AKIRA
JOHNSON Thomas
RAY Tuhina

MAITRE DE CONFERENCES DELEGUE

M. ROCHAT Jacques

Mécanique (IUT I)
Chimie (CUS)
Médecine interne
Thermodynamique
Automatique (IUT I)
Enzymologie
Analyse (IUT B) (Personnalité étrangère
habilitée à être directeur
de thèse).

Métallurgie (IUT I)
Géophysique-Glaciologie
Rhumatologie
Médecine Interne
Gynécologie et Obstétrique
Pédiatrie
Stomatologie
Bactériologie (Pharmacie)
Chimie-Physique
Mathématiques (IUT B) (Personnalité étran-
gère habilitée à être
directeur de thèse.)

Maths appliquées
Cancérologie
Cryptogamie
Anatomie Pathologie
Mécanique
Radiologie

Spectro Physique
Maths pures
Maths appliquées
Physique

Hygiène et Hydrologie (Pharmacie)

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Année universitaire 1978-1977

Président : M. Philippe TRAYNARD

Vice-présidents : M. René PAUTHENET

M. Georges LESPINARD

PROFESSEURS TITULAIRES

MM.	BENOIT Jean	Electronique - automatique
	BESSON Jean	Chimie minérale
	BLOCH Daniel	Physique du solide - cristallographie
	BONNETAIN Lucien	Génie chimique
	BONNIER Etienne	Métallurgie
*	BOUDOURIS Georges	Electronique - automatique
	BRISSONNEAU Pierre	Physique du solide - cristallographie
	BUYLE-BODIN Maurice	Electronique - automatique
	COUMES André	Electronique - automatique
	DURAND Francis	Métallurgie
	FELICI Noël	Electronique - automatique
	FOULARD Claude	Electronique - automatique
	LANCIA Roland	Electronique - automatique
	LONGEQUEUE Jean-Pierre	Physique nucléaire corpusculaire
	LESPINARD Georges	Mécanique
	MOREAU René	Mécanique
	PARIAUD Jean-Charles	Chimie - physique
	PAUTHENET René	Electronique - automatique
	PERRET René	Electronique - automatique
	POLOUJADOFF Michel	Electronique - automatique
	TRAYNARD Philippe	Chimie - physique
	VEILLON Gérard	Informatique fondamentale et appliquée
* en congé pour études		

PROFESSEURS SANS CHAIRE

MM.	BLIMAN Samuël	Electronique - automatique
	BOUVARD Maurice	Génie mécanique
	COHEN Joseph	Electronique - automatique
	GUYOT Pierre	Métallurgie physique
	LACOUME Jean-Louis	Electronique - automatique
	JOUBERT Jean-Claude	Physique du solide - cristallographie

.../...

MM.	ROBERT André	Chimie appliquée et des matériaux
	ROBERT François	Analyse numérique
	ZADWORNY François	Electronique - automatique

MAITRES DE CONFERENCES

MM.	ANCEAU François	Informatique fondamentale et appliquée
	CHARTIER Germain	Electronique - automatique
	CHIAVERINA Jean	Biologie, biochimie, agronomie
	IVANES Marcel	Electronique - automatique
	LESIEUR Marcel	Mécanique
	MORET Roger	Physique nucléaire - corpusculaire
	PIAU Jean-Michel	Mécanique
	PIERRARD Jean-Marie	Mécanique
	SABONNADIÈRE Jean-Claude	Informatique fondamentale et appliquée
Mme	SAUCIER Gabrielle	Informatique fondamentale et appliquée
M.	SOHM Jean-Claude	Chimie Physique

CHERCHEURS DU C.N.R.S. (Directeur et Maîtres de Recherche)

M.	FRUCHART Robert	Directeur de Recherche
MM.	ANSARA Ibrahim	Maître de Recherche
	BRONOEL Guy	Maître de Recherche
	CARRE René	Maître de Recherche
	DAVID René	Maître de Recherche
	DRIOLE Jean	Maître de Recherche
	KLEITZ Michel	Maître de Recherche
	LANDAU Ioan-Doré	Maître de Recherche
	MATHIEU Jean-Claude	Maître de Recherche
	MERMET Jean	Maître de Recherche
	MUNIER Jacques	Maître de Recherche

Personnalités habilitées à diriger des travaux de recherche (décision du Conseil Scientifique)

E.N.S.E.E.G.

MM.	BISCONDI Michel	Ecole des Mines St. Etienne (dépt. Métallurgie)
	BOOS Jean-Yves	Ecole des Mines St. Etienne (Métallurgie)
	DRIVER Julian	Ecole des Mines St. Etienne (Métallurgie)

.../...

MM. KOBYLANSKI André
 LE COZE Jean
 LESBATS Pierre
 LEVY Jacques
 RIEU Jean
 SAINFORT
 SOUQUET
 CAILLET Marcel
 COULON Michel
 GUILHOT Bernard
 LALAUZE René
 LANCELOT Francis
 SARRAZIN Pierre
 SOUSTELLE Michel
 THEVENOT François
 THOMAS Gérard
 TOUZAIN Philippe
 TRAN MINH Canh

Ecole des Mines St. Etienne (Métallurgie)
 Ecole des Mines St. Etienne (Métallurgie)
 Ecole des Mines St. Etienne (Métallurgie)
 Ecole des Mines St. Etienne (Métallurgie)
 Ecole des Mines St. Etienne (Métallurgie)
 C.E.N. Grenoble (Métallurgie)
 U.S.M.G.
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)
 Ecole des Mines St. Etienne (Chim. Min. Ph.)

E.N.S.E.R.G.

MM. BOREL
 KAMARINOS

Centre d'études nucléaires de Grenoble
 Centre national recherche scientifique

E.N.S.E.G.P.

M. BORNARD
 Mme CHERUY
 MM. DAVID
 DESCHIZEAUX

Centre national recherche scientifique
 Centre national recherche scientifique
 Centre national recherche scientifique
 Centre national recherche scientifique

*Je tiens à exprimer toute ma reconnaissance à
Monsieur Michel SAKAROVITCH de m'avoir accueilli dans son équipe et
m'avoir donné les moyens nécessaires à l'élaboration de cette thèse.*

*Je suis très sensible à l'honneur que me font
Monsieur Claude BENZAKEN et Monsieur Pierre-Jean LAURENT en acceptant
de participer à ce Jury.*

*Que Monsieur Jean FONLUPT' trouve ici le témoignage de ma
profonde gratitude pour sa bienveillance, sa disponibilité et ses
compétences. C'est grâce à ses critiques et ses encouragements que
ce travail a pu être réalisé.*

*Profitant de cette occasion, je voudrais exprimer mes cor-
diales amitiés à tous les membres de l'équipe de Recherche Opérationnelle
dont la sympathie et la sincérité me sont très chères, en particulier à
Monsieur Jean-Pierre UHRY qui m'a donné tant de conseils utiles.*

*Je remercie aussi Madame NEUMANN qui a assuré avec amabilité
et compétence la frappe de cette thèse, ainsi que Monsieur IGLESIAS
et son service de reprographie pour leur précieuse collaboration dans
la réalisation matérielle de ce travail.*

•

Kuon EA

TABLE DES MATIERES

	Pages
INTRODUCTION	1
 <u>CHAPITRE I - PRESENTATION DU PROBLEME ET FORMULATION MATHEMATIQUE</u>	
§1 - Présentation du problème	10
§2 - Relation avec le problème du voyageur de commerce	12
§3 - Formulation mathématique	13
§4 - Principe de résolution	20
 <u>CHAPITRE II - PROPRIETES DU PROGRAMME LINEAIRE (PC)</u>	
§1 - Dual du programme linéaire	22
§2 - Propriétés de la fonction w	25
 <u>CHAPITRE III - METHODES DE RESOLUTION DU PROBLEME (P')</u>	
§1 - Méthode ascendante	36
§2 - Méthode de pénalité	44
§3 - Méthode de sous-gradient généralisée	52
 <u>CHAPITRE IV - METHODES DE SEPARATION ET EVALUATION PROGRESSIVE</u>	
§1 - Introduction	67
§2 - Premier algorithme de SEP	67
§3 - Deuxième algorithme de SEP	81
§4 - Variation du paramètre m	83
 CONCLUSION	 87
 BIBLIOGRAPHIE	 91
 RAPPEL DES NOTATIONS	 94

INTRODUCTION

Comme l'indique son nom, le problème de ramassage scolaire est posé par le transport des écoliers dans une région rurale. C'est aussi le problème de transport des personnels ou des marchandises d'une entreprise. Il peut se formuler de la manière suivante : un responsable possède un certain nombre de véhicules. Il doit assurer le transport d'un certain nombre de personnes situées en des points $1, 2, \dots, n-1$ pour les ramener au point n (l'école ou l'usine). Il est évident qu'il ne peut pas surcharger un véhicule dont la capacité est limitée. Respectant cette contrainte, ce responsable souhaite soit minimiser le temps total passé par les usagers dans les cars, soit minimiser la somme des trajets de tous les cars. Il peut également envisager un critère plus "souple", intermédiaire aux deux critères de minimisation précédents, que nous décrirons plus tard. D'une manière très naturelle, ce problème se divise en deux phases distinctes : d'abord, le responsable en question doit essayer de regrouper les usagers, les personnes de même groupe sont affectées au même véhicule (éventuellement il doit fixer les points de départ de chaque car), ensuite, il lui reste à déterminer les trajets des cars en satisfaisant les critères d'optimisation qu'il a choisis.

D'une manière plus précise, ce problème, connu aussi sous le nom de problème de tournée, est caractérisé par quatre concepts :

(i) - Données : Elles peuvent être schématisées par un graphe $G = (\Omega, V)$ dans lequel chaque sommet $i \in \Omega = \{1, 2, \dots, n\}$ représente une localité (ville, commune ou zone) et chaque arc $e = (i, j)$ de V représente le plus court trajet (mesuré en kilomètres, en heures ou en francs) qui permet de passer de son origine i à son extrémité j . Soit c_{ij} le coût associé à ce trajet. Parmi les sommets de Ω , un seul est spécifié. C'est le centre n qui représente soit l'école, soit l'usine selon le cas.

(ii) - Contraintes : Les seules contraintes prises en compte ici sont les contraintes de capacité des véhicules (contraintes d'effectifs). Si le nombre de personnes à ramasser est supérieur à la capacité d'un car, on est obligé de partager Ω en plusieurs secteurs dont le nombre de personnes à ramasser ne dépasse pas la dite capacité. Ce partage est connu sous le nom de problème de sectorisation.

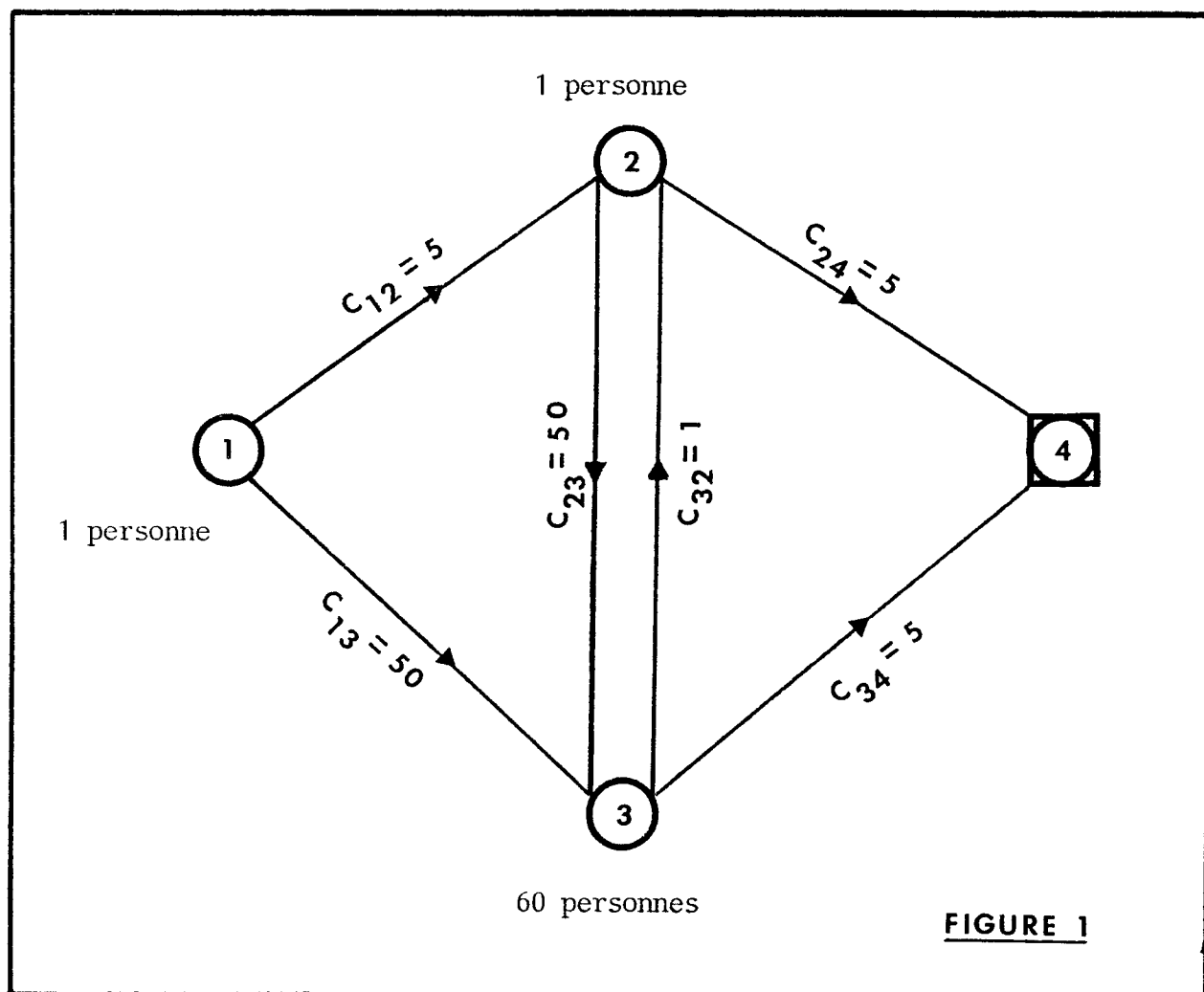
(iii) - Inconnues : Ce sont les parcours qui passent une et une seule fois sur chaque sommet sauf le centre n (emplacement de l'école ou de l'usine)

où ils peuvent passer plusieurs fois.

(iv) - Critères d'optimisation : On peut en envisager deux :

- a) Minimiser le temps total passé par les usagers dans les cars.
- b) Minimiser la distance totale du transport.

Ces deux critères pris indépendamment ne sont pas entièrement satisfaisants. Dans le premier cas, on risque de trop pénaliser les usagers individuellement. On trouve par exemple qu'il est préférable de faire attendre un usager isolé 50 minutes plutôt que de faire perdre à 60 usagers chacun une minute. Ceci n'est pas envisageable d'autant plus que les temps de parcours ne sont pas connus avec une précision suffisante. Ainsi, dans la figure 1, on trouve que le chemin (1,2,3,4) est optimal alors qu'il conviendrait de prendre le chemin (1,3,2,4)



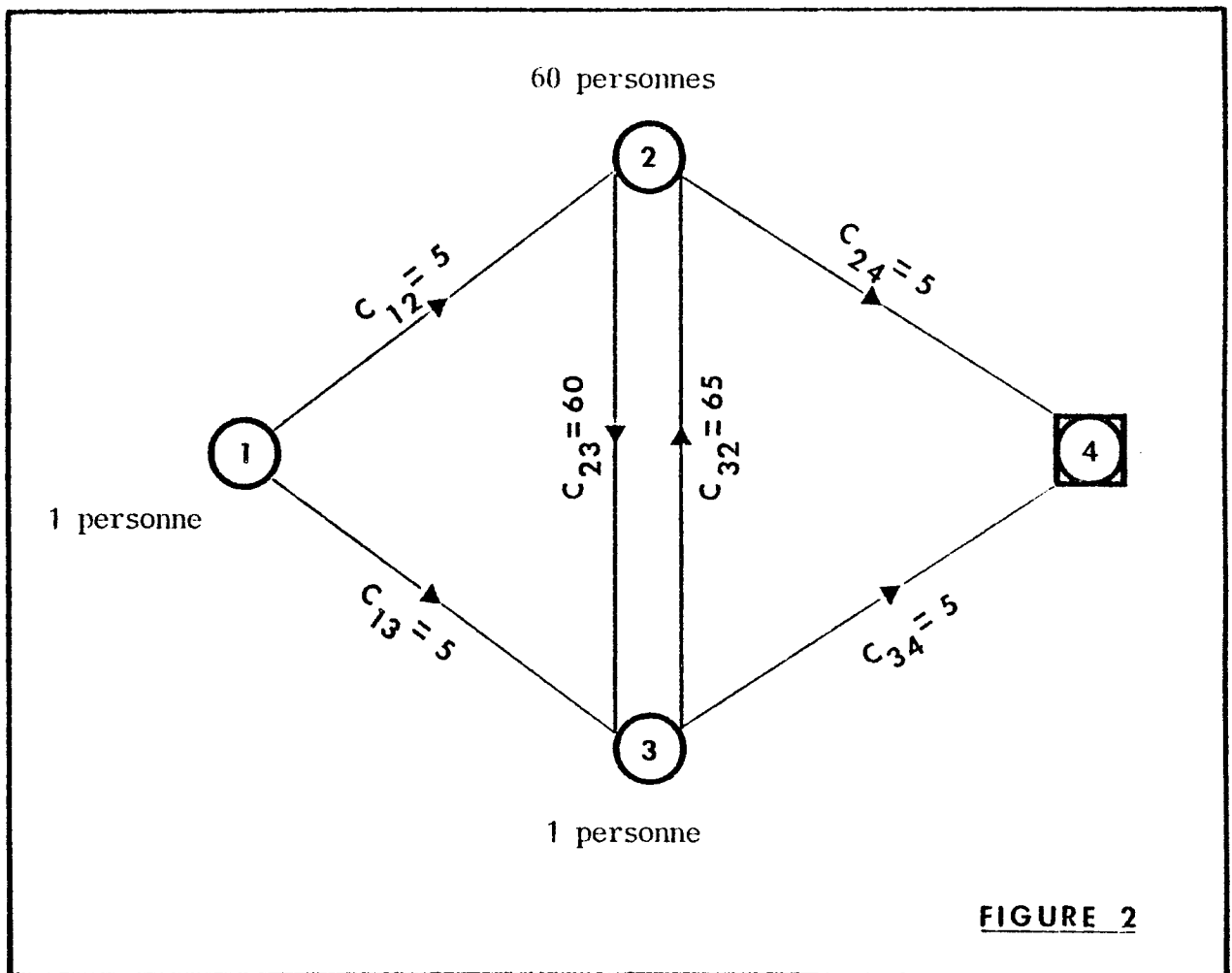
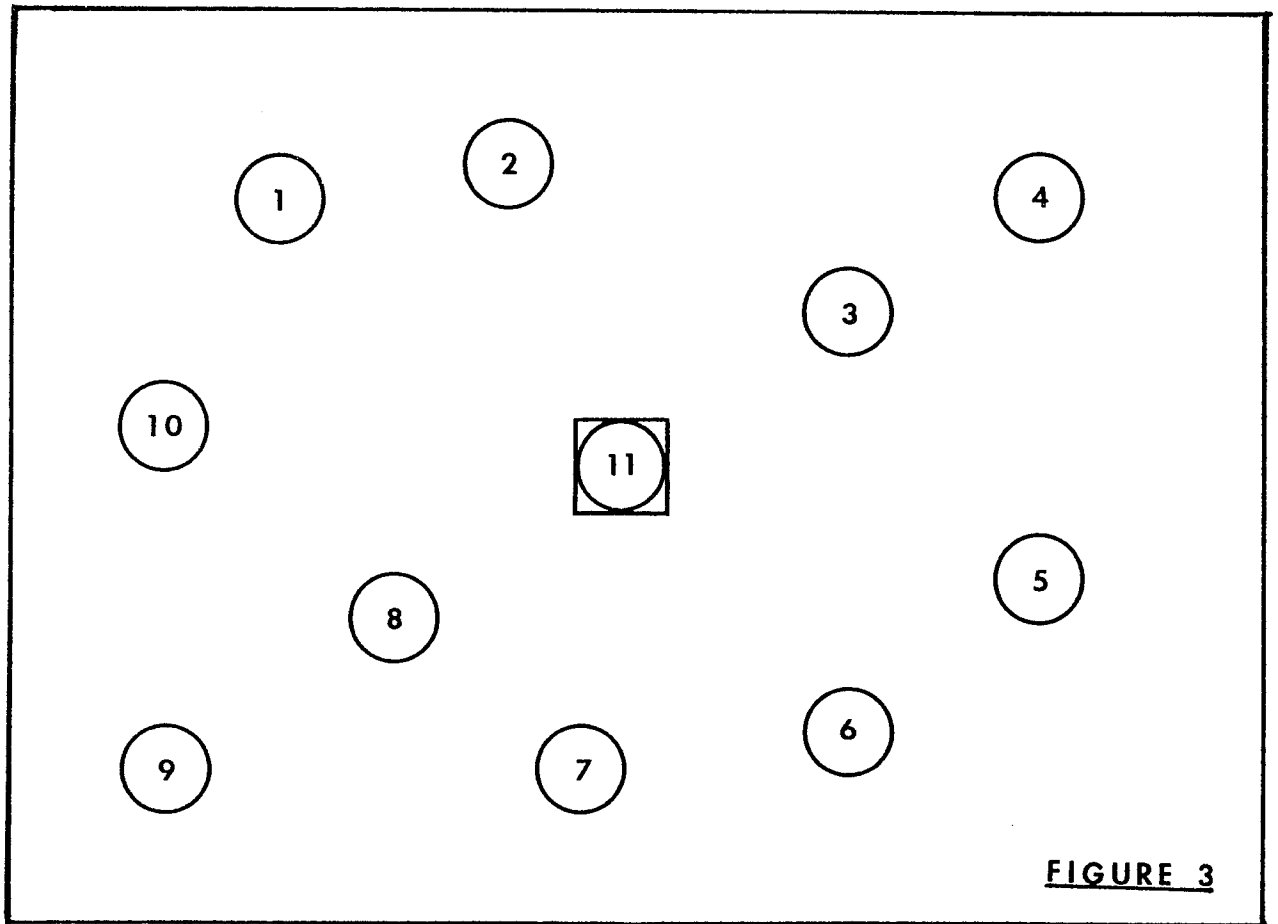


FIGURE 2

Dans le second cas, on risque de trop pénaliser les usagers en faisant faire de trop grands détours aux personnes qui se trouvent déjà dans le car alors qu'un trajet un peu plus long permettrait de passer d'abord aux points de ramassage où attendent une ou deux personnes. Ainsi, dans la figure 2, on trouve que le chemin (1,2,3,4) est optimal alors qu'il conviendrait de prendre le chemin (1,3,2,4).

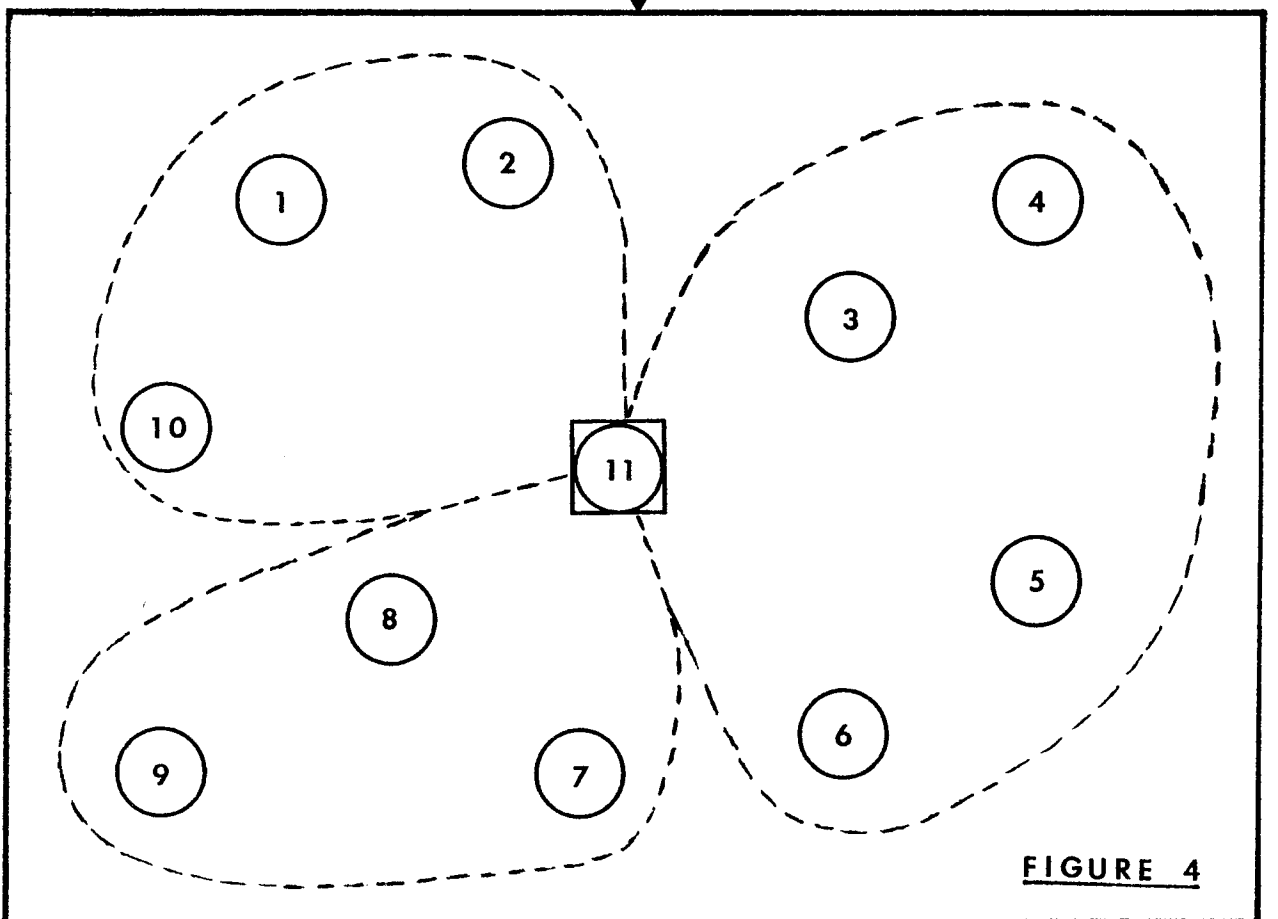
On verra plus tard comment on peut envisager un critère plus souple qui tient compte des deux critères précédents.

Le schéma constitué par les figures 3, 4, 5 illustre bien les deux phases d'organisation des circuits de ramassage.



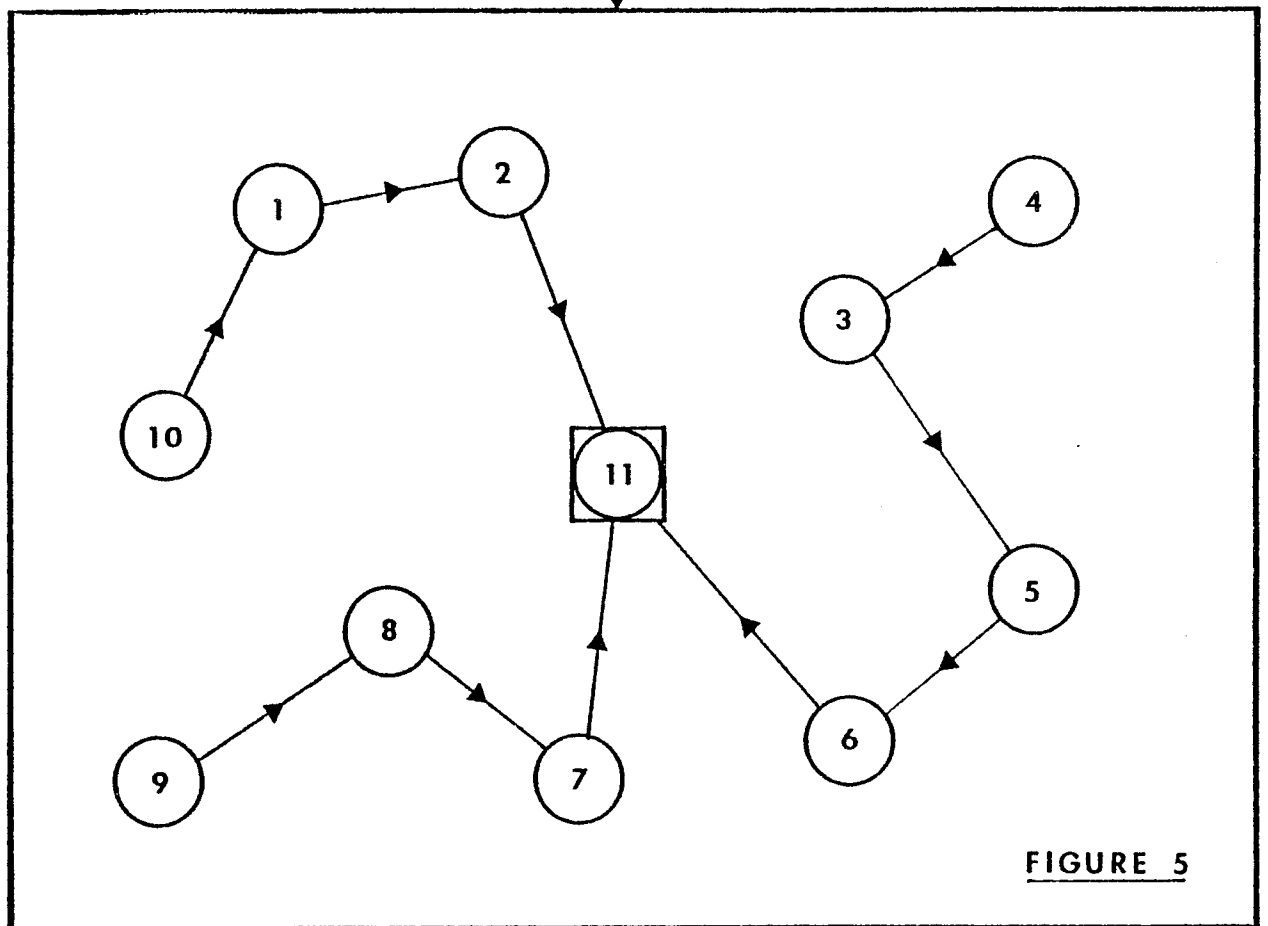
Première partie :

Sectorisation



Deuxième partie :

Détermination des meilleurs trajets



Le problème que nous allons étudier représente la deuxième partie (ou le problème global si le nombre total des usagers ne dépasse pas la capacité d'un véhicule). D'une manière plus précise, un car part d'une origine donnée, le point 1, doit ramasser un certain nombre de personnes $m_1, m_2, \dots, m_{n-1}$, situées en des points 1, 2, ..., $n-1$. La distance entre deux points i et j est donnée par c_{ij} (qui peut être différente de c_{ji} à cause d'éventuels sens interdits). Le temps que met le car pour aller de i à j est supposé proportionnel à cette distance. La figure 6 illustre bien un exemple où $n = 5$. Comme critère d'optimisation, nous nous proposons de prendre le critère (iv)-a.

Si au lieu de prendre (iv)-a comme critère de minimisation, nous nous intéressons à déterminer une tournée qui minimise la distance totale du parcours, nous retrouvons alors une version du célèbre problème du voyageur de commerce [2], [10].

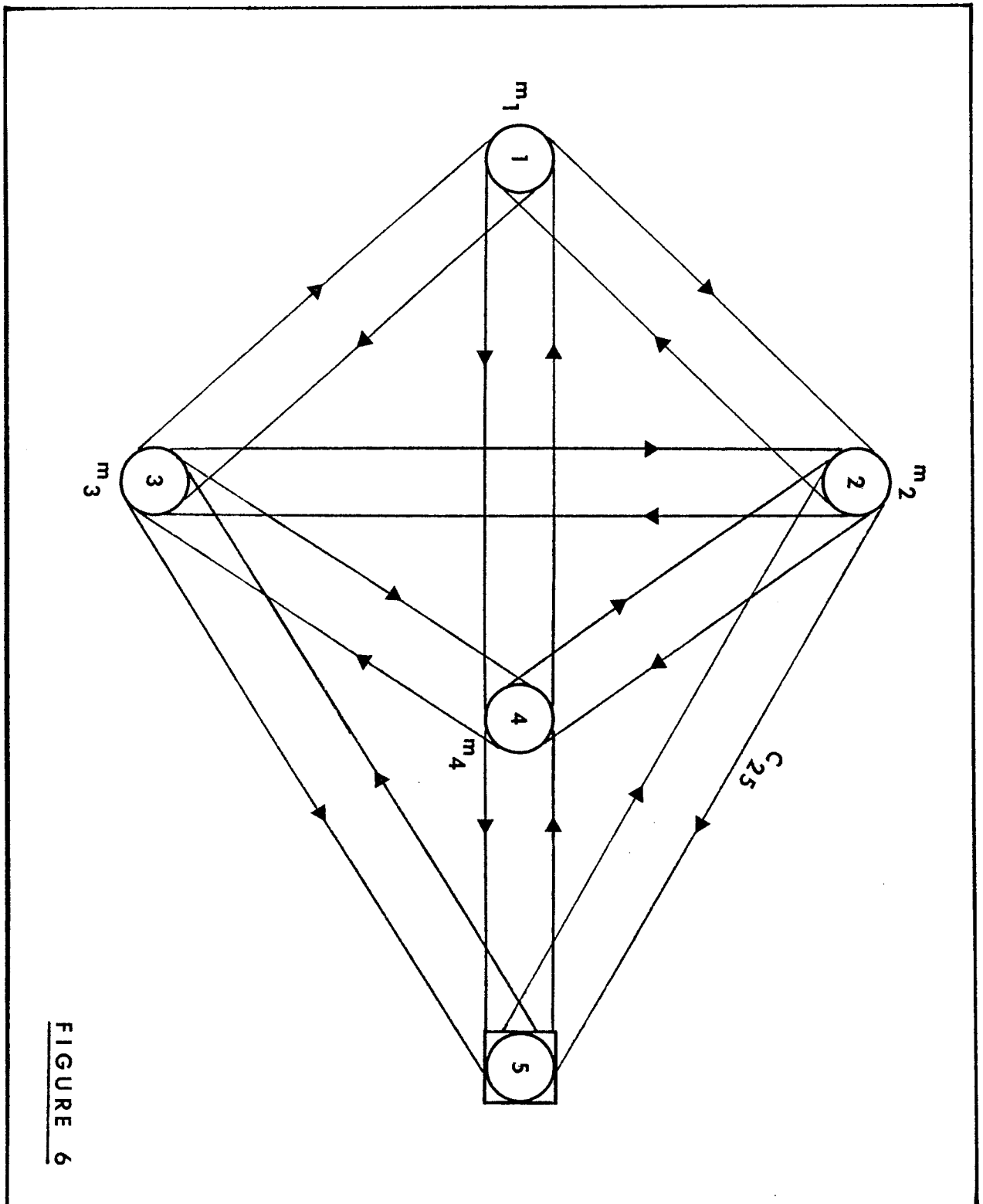


FIGURE 6

Bien que le critère d'optimisation choisi soit (iv)-a, on verra au chapitre I que, plus m_1 est grand, plus on s'approche du problème du voyageur de commerce, c'est à dire plus on a tendance à minimiser la longueur totale du parcours. Par conséquent, m_1 peut aussi jouer le rôle de pondération entre les deux critères a et b de (iv).

Le travail que nous allons entreprendre pour résoudre le problème posé consiste à étudier un programme linéaire continu dont la solution optimale constitue une borne inférieure de la tournée optimale et, ensuite, mettre en oeuvre une procédure de Séparation et Evaluation Progressive (Branch and Bound) pour déterminer complètement cette tournée optimale. Il comporte quatre chapitres. Le premier sera consacré à la modélisation du problème et à établir son rapport avec le problème du voyageur de commerce. On construira ainsi un programme linéaire continu (PC) dont les propriétés seront présentées au chapitre II. Dans le chapitre III, on trouvera une comparaison des différentes méthodes utilisées pour résoudre (PC). La meilleure sera utilisée par le Branch and Bound dont deux procédures différentes seront présentées dans le chapitre IV.

Pour décrire tout cela, nous avons besoin de quelques définitions et résultats de la théorie des graphes. Les lecteurs peu familiers avec ces notions peuvent les trouver dans [1] ou [15].

CHAPITRE I

PRESENTATION DU PROBLEME ET FORMULATION MATHEMATIQUE

§1 - PRESENTATION DU PROBLEME

Soient $G = (X, E)$ un graphe simple, orienté, dont $X = \{1, 2, \dots, n\}$ est l'ensemble des sommets, $E = \{(i, j) / i \neq j, i \in \{2, \dots, n-1\}, j \in \{2, \dots, n-1\}\} \cup \{(1, j) / j \in \{2, \dots, n-1\}\} \cup \{(i, n) / i \in \{2, \dots, n-1\}\}$; c un système de coût réel sur les arcs. Sur chaque sommet $i \in \{1, 2, \dots, n-1\}$ se trouvent m_i personnes (voir figure 6 de l'Introduction).

Définition 1 : On appelle chemin hamiltonien du sommet 1 au sommet n , un chemin de 1 à n qui passe une et une seule fois par tous les sommets $2, 3, \dots, n-1$.

Remarque 1 : Il est facile de voir qu'un chemin hamiltonien de 1 à n contient $n-1$ arcs et qu'il existe une correspondance bijective entre l'ensemble des chemins hamiltoniens de 1 à n et l'ensemble $\mathcal{S}_{[2, n-1]}$ des permutations de l'ensemble $\{2, 3, \dots, n-1\}$. Au chemin hamiltonien $H = (e_1, e_2, \dots, e_{n-1})$ de 1 à n correspond la permutation $\sigma = (\sigma_2, \sigma_3, \dots, \sigma_{n-1}) \in \mathcal{S}_{[2, n-1]}$, où σ_i est l'extrémité terminale de l'arc e_{i-1} et réciproquement.

Définition 2 : On appelle coût d'un chemin hamiltonien $H = (e_1, \dots, e_{n-1})$, la somme

$$c(H) = \sum_{i=1}^{n-1} c(e_i) \text{ des coûts des arcs de } H.$$

Notations 1 : Dans toute la suite, sauf indications particulières, nous notons :

- (i) $c(e)$ ou c_{ij} le coût de l'arc $e = (i, j)$
- (ii) $c(H)$ le coût du chemin hamiltonien H de 1 à n .
- (iii) $H = \{(H, c(H)) / H \in \text{l'ensemble des chemins hamiltoniens de 1 à } n\}$ et tout élément de H est appelé tout simplement chemin hamiltonien.
- (iv) $\mathcal{S}_{[2, n-1]}$ l'ensemble des permutations de $\{2, 3, \dots, n-1\}$.

Définition 3 : Un chemin hamiltonien $(H_0, c(H_0))$ est dit optimal si

$$\forall (H, c(H)) \in H \quad c(H_0) \leq c(H).$$

Une version du célèbre problème du voyageur de commerce est la recherche d'un chemin hamiltonien optimal [2], [5], [6], [10], [17]. C'est un

problème "difficile" du point de vue de complexité algorithmique, appartenant à la classe des problèmes dits NP-complets, c'est à dire à la classe des problèmes tels que, si l'on pouvait les résoudre par un algorithme polynomial, tous les problèmes combinatoires classiques seraient résolus par des algorithmes polynomiaux [8].

Définition 4 : Soit $(m_1, m_2, \dots, m_{n-1}) \in \mathbb{N}^{n-1}$. On appelle tournée par rapport au système $(m_1, m_2, \dots, m_{n-1})$ -ou tout simplement tournée si aucune confusion n'est à craindre- le couple $(T, c(T))$ où T est un chemin hamiltonien $(e_1, e_2, \dots, e_{n-1})$ de 1 à n et où $c(T)$ est donné par :

$$c(T) = m_1 \cdot c(e_1) + (m_1 + m_2) \cdot c(e_2) + \dots + (m_1 + m_2 + \dots + m_{n-1}) \cdot c(e_{n-1}) \quad (1)$$

Désignons par $\mathcal{T}$ l'ensemble des tournées.

Définition 5 : Une tournée T_0 est dite optimale si

$$\forall (T, c(T)) \in \mathcal{T}, \quad c(T_0) \leq c(T).$$

Nous nous proposons d'étudier le problème de recherche d'une tournée optimale.

Remarque 2 : On peut toujours supposer que $m_1 = 1$ pour $i = 2, 3, \dots, n-1$, sinon on remplace le sommet i en m_i sommets $i_1, i_2, \dots, i_{m_i}$ et l'on pose :

$$c_{i_k i_\ell} = 0 \quad \text{pour } k, \ell \in \{1, 2, \dots, m_i\} \text{ et } k \neq \ell$$

$$c_{i_k j} = c_{ij} \quad \text{pour } k \in \{1, 2, \dots, m_i\} \text{ et } j \neq i, j \in \{2, 3, \dots, n\}$$

De plus, on suppose que $m_1 = m+1$ (un conducteur et m personnes éventuelle au départ). La relation (1) devient donc :

$$c(T) = (m+1) \cdot c(e_1) + (m+2) \cdot c(e_2) + \dots + (m+n-1) \cdot c(e_{n-1}) \quad (2)$$

C'est ainsi que nous sommes conduits à étudier le problème

(P) : Comment déterminer une tournée optimale par rapport au système

$$\underbrace{(m+1, 1, \dots, 1)}_{(n-1) \text{ composantes}}$$

Remarque 3 : Il est facile de voir que la solution du problème (P) ne change pas si l'on ajoute une même constante à tous les coûts des arcs. On peut donc supposer dans la suite que le coût c est non négatif.

§2 - RELATION AVEC LE PROBLEME DU VOYAGEUR DE COMMERCE

Intuitivement, on sent bien que, plus la valeur de m augmente, plus l'effet des $n-2$ personnes placées aux points $2, 3, \dots, n-1$ diminue. Donc, plus m est grand, plus on se ramène au cas où $m_1 = 1$ et $m_i = 0$ pour $i = 2, 3, \dots, n-1$ ou encore plus le problème (P) se ramène au problème du voyageur de commerce (avec les coûts multipliés par la constante $m+1$). Le théorème suivant confirme cette intuition.

Théorème 1 : Le problème (P) est NP-complet.

Démonstration : Pour simplifier la démonstration, on peut supposer que le coût c est entier.

Soit m entier $= 1 + \frac{n(n-1)}{2} \cdot \max_{e \in E} c(e)$ et soient :

- (i) $(H_0, c(H_0))$ un chemin hamiltonien optimal.
- (ii) $(T, c(T))$ la tournée qui correspond à $(H_0, c(H_0))$, c'est à dire que H_0 et T désignent le même chemin hamiltonien de 1 à n .
- (iii) $(T_0, c(T_0))$ une tournée optimale.
- (iv) $(H, c(H))$ le chemin hamiltonien qui correspond à $(T_0, c(T_0))$, c'est à dire que T_0 et H désignent le même chemin hamiltonien de 1 à n .

Remarquons que, si $T_1 = (e_1, e_2, \dots, e_{n-1})$ désigne un chemin hamiltonien quelconque de 1 à n (en tant que tournée) et H_1 désigne son correspondant en tant que chemin hamiltonien tout simplement, nous avons :

$$\begin{aligned} c(T_1) &= (m+1) \cdot c(e_1) + (m+2) \cdot c(e_2) + \dots + (m+n-1) \cdot c(e_{n-1}) \\ &= m \cdot c(H_1) + c'(T_1) \end{aligned} \tag{3}$$

où $c'(T_1) = c(e_1) + 2 \cdot c(e_2) + \dots + (n-1) \cdot c(e_{n-1})$

Pour terminer la démonstration, il suffit de montrer que le chemin hamiltonien $(H, c(H))$ qui correspond à la tournée optimale $(T_O, c(T_O))$ est aussi optimal. Supposons le contraire, on aurait, puisque c est entier :

$$c(H) - c(H_O) \geq 1$$

D'autre part

$$\begin{aligned} c(T_O) - c(T) &= c'(T_O) + m.c(H) - c'(T) - m.c(H_O) \\ &= (c'(T_O) - c'(T)) + m(c(H) - c(H_O)) \end{aligned}$$

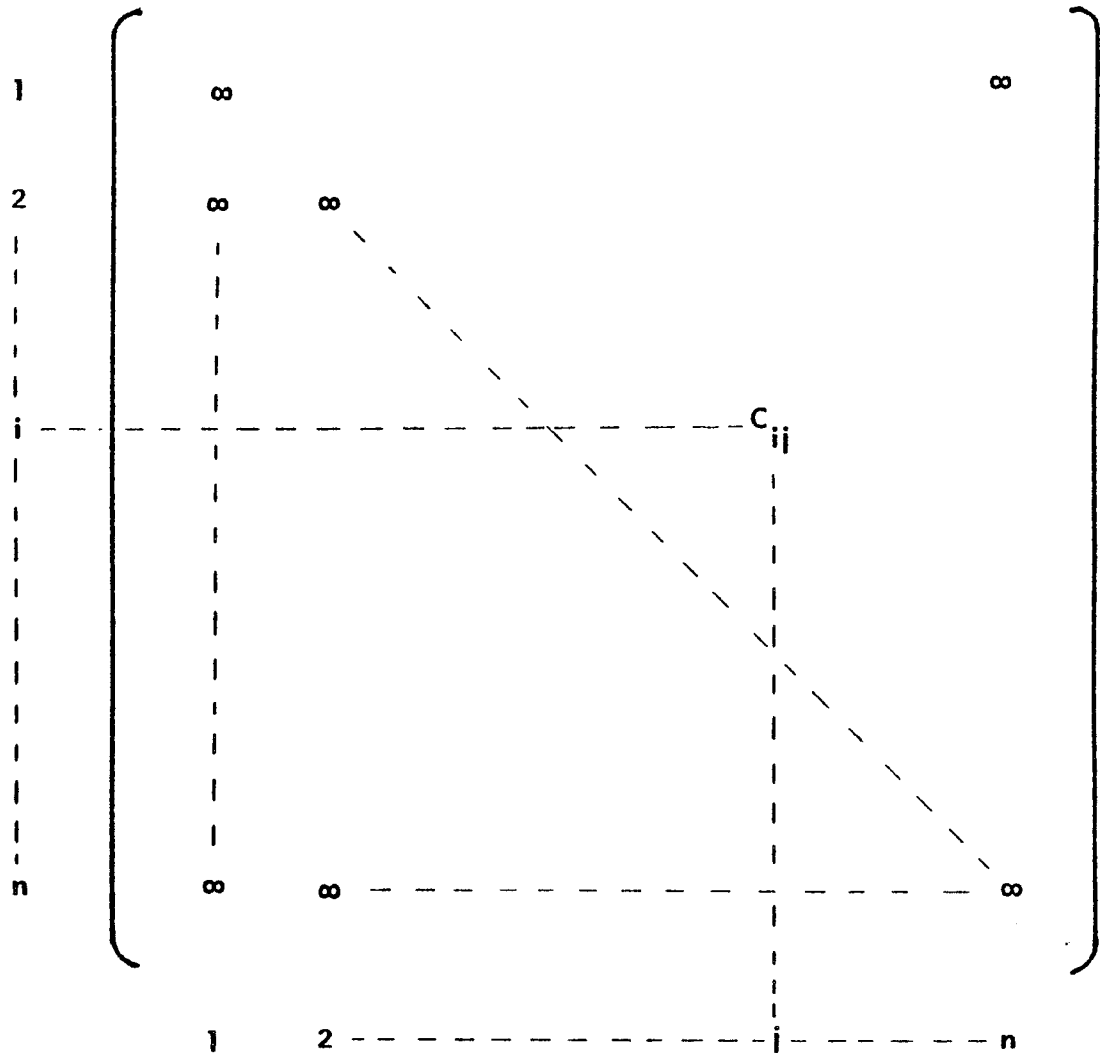
Le choix de m implique $c(T_O) - c(T) > 0$, ce qui contredit à l'optimalité de T_O .

Donc, pour m assez grand, la solution du problème (P) est aussi celle du problème du voyageur de commerce.

Ce théorème fait perdre tout espoir de trouver un algorithme polynomial pour résoudre le problème (P) ; il montre clairement que le problème (P) est au moins aussi difficile -tant sur le plan de complexité algorithmique que sur le plan de résolution pratique- que le problème du voyageur de commerce. Nous verrons au chapitre IV que, en pratique, (P) est bien plus difficile que le problème du voyageur de commerce.

§3 - FORMULATION MATHÉMATIQUE

Pour simplifier l'exposé et les calculs pratiques, nous supposons dans la suite que le graphe $G = (X, E)$ est simple, orienté, complet et possède des boucles, $X = \{1, 2, \dots, n\}$ et $E = \{(i, j) / i \in X, j \in X\}$. Nous mettons les coûts infinis sur les boucles, sur les arcs $(i, 1)$ pour $i = 2, 3, \dots, n$, sur les arcs (n, i) pour $i = 1, 2, \dots, n-1$ et aussi sur l'arc $(1, n)$. La matrice des coûts prend alors la forme suivante :



3.1 - Formulation

Dans cette partie, on construit un programme linéaire en nombres entiers admettant les tournées comme solutions réalisables. Ce programme possède 2 catégories de variables, les variables y_{ij} dites de positions et les variables x_{ij} dites de trafics.

Variables de positions : A chaque couple de sommets (i,j) , $i = 2,3,\dots,n-1$, $j = 2,3,\dots,n-1$, nous faisons correspondre une variable bivalente y_{ij} égale à 0 ou 1 ; $y_{ij} = 1$ signifie que le sommet i est placé en j ème position, $y_{ij} = 0$ sinon. Le vecteur y dont les composantes sont y_{ij} appartient donc à $\{0,1\}^{(n-2) \times (n-2)}$

Les contraintes :

$$\sum_{j=2}^{n-1} y_{ij} = 1, \quad i = 2, 3, \dots, n-1 \quad (4)$$

$$\sum_{i=2}^{n-1} y_{ij} = 1, \quad j = 2, 3, \dots, n-1 \quad (5)$$

définissent ainsi une permutation des sommets $2, 3, \dots, n-1$. Une telle permutation étant définie, le trafic total entrant dans le sommet i , c'est à dire le nombre total de personnes entrant dans le sommet i avec le car est :

$$m + \sum_{j=2}^{n-1} (j-1) \cdot y_{ij}, \quad i = 2, 3, \dots, n-1 \quad (6)$$

Le trafic total entrant dans le sommet n est $m+n-1$.

De même, le trafic total sortant du sommet i est :

$$m + \sum_{j=2}^{n-1} j \cdot y_{ij}, \quad i = 2, 3, \dots, n-1 \quad (7)$$

Le trafic total sortant du sommet 1 est $m+1$.

Les trafics entrant dans le sommet 1 et sortant du sommet n sont nuls.

Variables de trafics : A chaque arc (i, j) du graphe G , $i = 1, 2, \dots, n, j = 1, 2, \dots, n$, nous associons une variable entière non négative x_{ij} qui représente le trafic passant du sommet i au sommet j . Le vecteur x dont les composantes sont x_{ij} appartient donc à $N^{n \times n}$.

En termes de variables x_{ij} , le trafic total entrant dans le sommet j est :

$$\sum_{i=1}^n x_{ij}, \quad j = 1, 2, \dots, n \quad (8)$$

Le trafic total sortant du sommet i est :

$$\sum_{j=1}^n x_{ij}, \quad i = 1, 2, \dots, n \quad (9)$$

Ces considérations nous conduisent au programme linéaire en nombres entiers suivant :

$$\begin{aligned}
 & y_{ij} = 0 \text{ ou } 1 \quad i = 2, 3, \dots, n-1 \quad j = 2, 3, \dots, n-1 \\
 & x_{ij} = \text{entier} \geq 0 \quad i = 1, 2, \dots, n \quad j = 1, 2, \dots, n. \\
 & \sum_{j=2}^{n-1} y_{ij} = 1 \quad i = 2, 3, \dots, n-1 \\
 & \sum_{i=2}^{n-1} y_{ij} = 1 \quad j = 2, 3, \dots, n-1 \\
 & \sum_{j=1}^n x_{1j} = m+1 \\
 & \sum_{j=1}^n x_{ij} = m + \sum_{j=2}^{n-1} j \cdot y_{ij} \quad i = 2, 3, \dots, n-1 \\
 & \sum_{j=1}^n x_{nj} = 0 \\
 & \sum_{i=1}^n x_{il} = 0 \\
 & \sum_{i=1}^n x_{ij} = m + \sum_{i=2}^{n-1} (i-1) \cdot y_{ji} \quad j = 2, 3, \dots, n-1 \\
 & \sum_{i=1}^n x_{in} = m+n-1 \\
 & \sum_{i,j} c_{ij} \cdot x_{ij} = z(\min)
 \end{aligned}
 \quad (PLE)$$

Remarque 4 :

Il est évident que, pour une tournée donnée, la permutation des sommets $2, 3, \dots, n-1$ qui lui correspond vérifie les contraintes (4) et (5). D'autre part, les trafics sur les arcs correspondants à la même tournée vérifient les autres contraintes de (PLE). Donc une tournée est bien une solution réalisable de (PLE).

Remarque 5 :

Rappelons nous que le problème du voyageur de commerce n'est autre que le problème de tournée par rapport au système $(1, 0, 0, \dots, 0)$. Dans ce cas, $(n-1)$ composantes, puisqu'il n'y a personne sur les sommets intermédiaires $2, 3, \dots, n-1$, les variables

de positions y_{ij} servant à définir les trafics sortant et entrant dans les sommets seront enlevées. D'autre part, puisque les trafics qui passent par tous les sommets sont tous égaux à 1, la contrainte $x \in \mathbb{N}^{n \times n}$ sera remplacée par $x \in \{0,1\}^{n \times n}$.

Le programme (PLE) devient alors :

$$(PLE') \left\{ \begin{array}{ll} x_{ij} = 0 \text{ ou } 1 & i = 1, 2, \dots, n \quad j = 1, 2, \dots, n \\ \sum_{j=1}^n x_{ij} = 1 & i = 1, 2, \dots, n-1 \\ \sum_{j=1}^n x_{nj} = 0 \\ \sum_{i=1}^n x_{ij} = 1 & j = 2, 3, \dots, n \\ \sum_{i=1}^n x_{i1} = 0 \\ \sum_{i,j} c_{ij} x_{ij} = z(\min) \end{array} \right.$$

Bien qu'il s'agisse ici d'un chemin hamiltonien, on reconnaît tout de suite la formulation du problème du voyageur de commerce comme problème d'affectation utilisée dans [17].

Si on considère le graphe partiel défini par les arcs (i,j) tels que $x_{ij} = 1$, où x est une solution réalisable de (PLE') , on peut imaginer que ce graphe peut ne pas être connexe comme illustre la figure 1. Par contre, si x est une solution réalisable de (PLE) , le graphe partiel correspondant est nécessairement connexe ; ceci vient du fait qu'on suppose que pour chaque sommet du graphe, le trafic qui sort est supérieur d'une unité au trafic qui entre.

Notons enfin que (PLE) est un programme en nombres entiers et que (PLE') est en variables bivalentes donc que (PLE) est plus difficile que (PLE') .

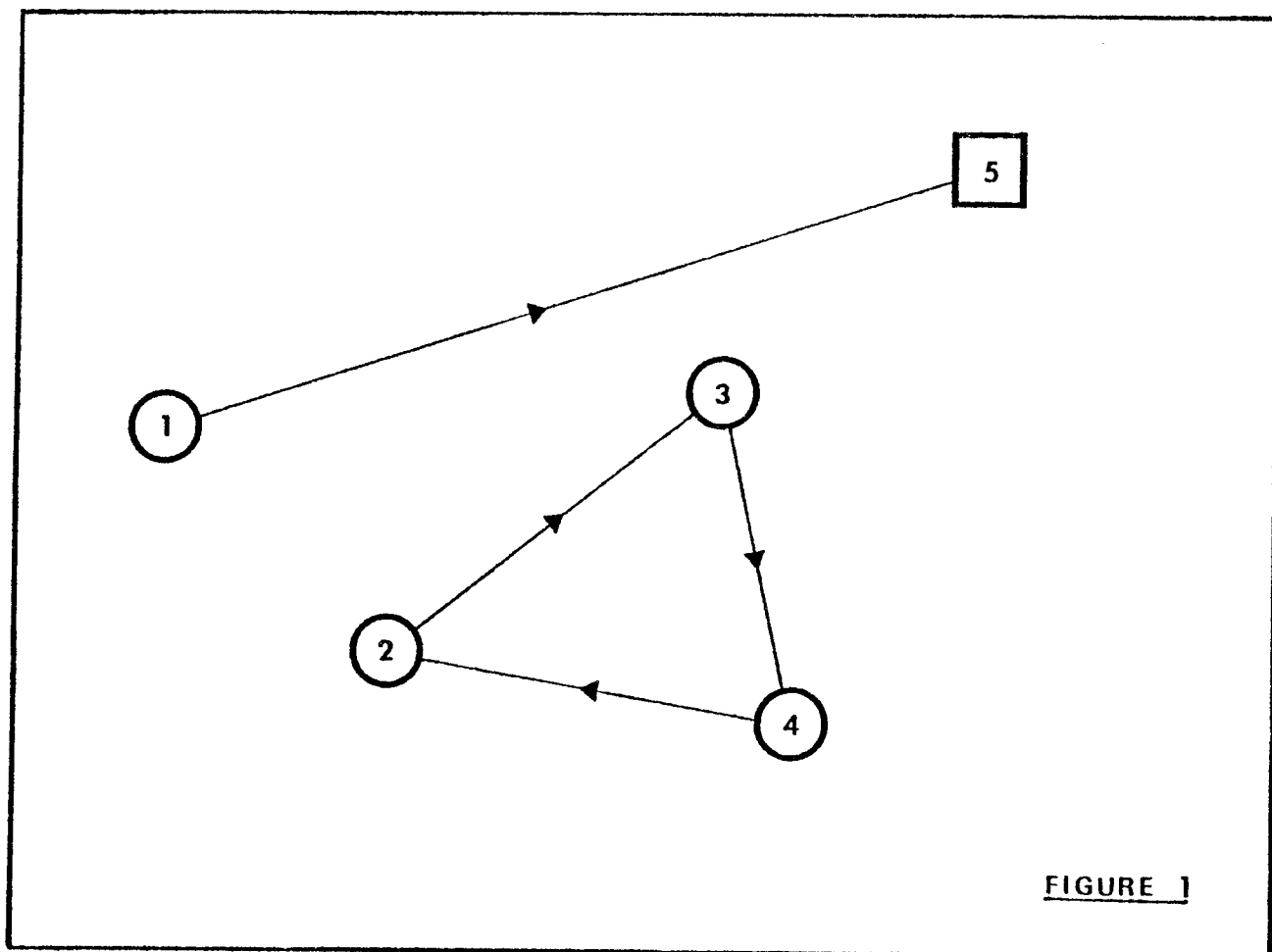


FIGURE 1

3.2 - Tournées et solutions réalisables des programmes linéaires

Si on remplace les contraintes $y_{ij} = 0$ ou 1 et $x_{ij} \in \mathbb{N}$ de (PLE) par $y_{ij} \geq 0$ et $x_{ij} \geq 0$, on obtient un programme linéaire continu appelé dans toute la suite le programme (PC). Nous avons vu (Remarque 4) qu'une tournée est une solution réalisable de (PLE), donc à fortiori, elle est une solution réalisable de (PC). Par conséquent, la valeur de la solution de (PC) à l'optimum constitue une borne inférieure des coûts de toutes les tournées. Cependant, les points suivants méritent notre attention.

Remarque 6 :

Une solution réalisable de (PLE) n'est pas nécessairement une tournée comme le montre l'exemple suivant (figure 2) :

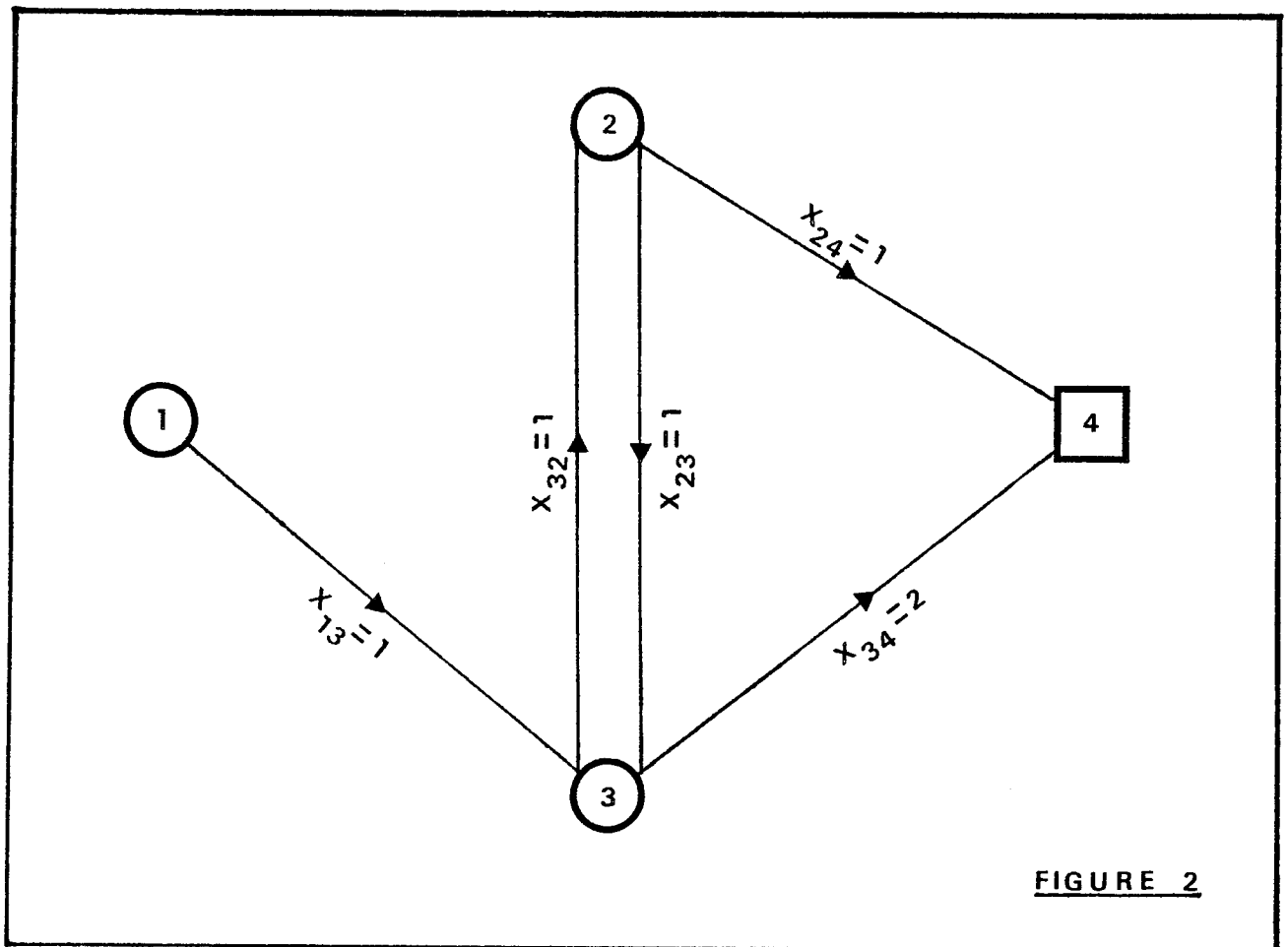


FIGURE 2

$$n = 4, m = 0,$$

$$c_{ij} = 0 \text{ sur les arcs marqués, } c_{ij} = \infty \text{ sur les arcs non marqués}$$

$$Y_{22} = Y_{33} = 1, Y_{ij} = 0 \text{ ailleurs}$$

$$x_{13} = x_{23} = x_{32} = x_{24} = 1, x_{34} = 2$$

La solution (y,x) ainsi définie est réalisable (et aussi optimale) de (PLE), mais ce n'est pas une tournée à cause du circuit $(2,3,2)$.

Remarque 7 :

La valeur de la solution optimale du programme (PC) peut être différente du coût de la tournée optimale. Reprenons le même exemple (figure 2) dans lequel le coût c est modifié comme suit :

$$c_{13} = c_{23} = c_{24} = c_{34} = 1, c_{32} = 3, c_{ij} = \infty \text{ sur les arcs non marqués.}$$

Il n'y a qu'une seule tournée de coût non égal à ∞ , c'est $T_0 = (1,3,2,4)$.

Elle est donc optimale et son coût est $c(T_0) = 1.1 + 2.3 + 3.1 = 10$.

D'autre part, la même solution (y,x) que dans la remarque 6 est réalisable dans (PC) et la valeur de la fonction objective correspondante est :

$$z = 1.1 + 1.3 + 1.1 + 2.1 + 1.1 = 8 < 10$$

Donc la valeur de la solution optimale de (PC) est strictement inférieure au coût de la tournée optimale .

Signalons enfin que (PC) peut admettre aussi des solutions optimales non entières.

§4 - PRINCIPE DE RESOLUTION

Nous avons vu que le problème (P) est NP-complet et qu'il permet de traiter en même temps le problème du voyageur de commerce. Pour le résoudre, nous considérons d'abord le programme linéaire en nombres entiers (PLE), puis en oubliant les contraintes d'intégrité de (PLE), le programme linéaire (PC) qui fournit une borne inférieure du coût de la tournée optimale. Nous nous servons ensuite de cette borne pour mettre en oeuvre une procédure de séparation et évaluation progressive qui permet de résoudre complètement le problème (P).

Cette façon de procéder constitue une des méthodes les plus efficaces en optimisation combinatoire et est fréquemment utilisée, surtout pour résoudre les problèmes NP-complets. Held et Karp ont ainsi fait un grand pas dans l'histoire du problème du voyageur de commerce [5], [6]. Ils ont réussi à résoudre ce problème dont les tailles dépassent largement celles qu'on pouvait traiter à l'époque de Little [10]. Plus récemment, les travaux de Fargier [4] sur l'implantation d'infrastructure dans un problème de transport, les travaux de Mukendi [11] sur le problème de m-centre utilisent, eux aussi, cette technique de manière très efficace.

Le programme linéaire continu (PC) est en général de très grande taille ($4n-4$ contraintes et $2n^2-2n+4$ variables) et n'est guère passible de la méthode classique du simplexe. D'autre part, la structure même de ce programme linéaire, ainsi que quelques unes de ses propriétés que nous donnerons au Chapitre II, nous conduisent à recourir à des méthodes de décomposition du type Benders [3], ou plus généralement à des méthodes de sous-gradient [6], [19].

CHAPITRE II

PROPRIETES DU PROGRAMME LINEAIRE (PC)

§1 - DUAL DU PROGRAMME LINEAIRE

Puisque notre but principal est d'approcher inférieurement le coût de la tournée optimale par la valeur de la solution optimale du programme linéaire continu (PC), et que la fonction objective de (PC) est à minimiser il est naturel d'étudier, non pas (PC) lui-même, mais son dual [14]. De cette manière, même si la méthode utilisée pour le résoudre n'est pas exacte, même si l'on arrête l'opération sans arriver à l'optimum, tant que l'on a une solution réalisable du programme dual, on aura toujours une borne inférieure du coût de la tournée optimale.

Rappelons d'abord que (PC) n'est autre que le programme (PLE) dans lequel on remplace les contraintes d'intégrité de y et x par $y \geq 0$ et $x \geq 0$. Sous forme abrégée, le programme (PC) s'écrit :

$$(PC) \begin{cases} y \geq 0 \\ x \geq 0 \\ A1.y = 1 \\ -B.y + A2.x = M \\ c.x = z \text{ (min)} \end{cases}$$

où $y \in \mathbb{R}^{(n-2) \cdot (n-2)}$, $1 \in \mathbb{R}^{2 \cdot (n-2)}$, $x \in \mathbb{R}^{n \cdot n}$, $M \in \mathbb{R}^{2 \cdot n}$, $t_c \in \mathbb{R}^{n \cdot n}$,

$A1$ d'ordre $2 \cdot (n-2) \times (n-2)$ (matrice de $2 \cdot (n-2)$ lignes et de $(n-2) \cdot (n-2)$ colonnes),

B d'ordre $2 \cdot n \times (n-2) \cdot (n-2)$,

$A2$ d'ordre $2 \cdot n \times n \cdot n$

sont de forme :

$$y = \begin{bmatrix} y_{22} \\ y_{23} \\ \vdots \\ y_{2(n-1)} \\ y_{32} \\ y_{33} \\ \vdots \\ y_{3(n-1)} \\ \vdots \\ y_{(n-1)2} \\ y_{(n-1)3} \\ \vdots \\ y_{(n-1)(n-1)} \end{bmatrix}, \quad x = \begin{bmatrix} x_{11} \\ x_{12} \\ \vdots \\ x_{1n} \\ x_{21} \\ x_{22} \\ \vdots \\ x_{2n} \\ \vdots \\ x_{n1} \\ x_{n2} \\ \vdots \\ x_{nn} \end{bmatrix}, \quad t_c = \begin{bmatrix} c_{11} \\ c_{12} \\ \vdots \\ c_{1n} \\ c_{21} \\ c_{22} \\ \vdots \\ c_{2n} \\ \vdots \\ c_{n1} \\ c_{n2} \\ \vdots \\ c_{nn} \end{bmatrix}$$

$$\mathbf{1} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \begin{matrix} \uparrow \\ 2 \cdot (n-2) \\ \downarrow \end{matrix}, \quad M = \begin{bmatrix} m+1 \\ m \\ \vdots \\ m \\ 0 \\ 0 \\ m \\ \vdots \\ m \\ m+n-1 \end{bmatrix} \begin{matrix} \uparrow \\ 2 \cdot n \\ \downarrow \end{matrix}$$

[illegible]

La matrice A2 a exactement la même forme que A1 sauf dans le marquage des dimensions, on a le nombre n partout au lieu de n-2 comme dans A1. Ce sont les matrice des contraintes du problème d'affectation [17], de taille (n-2) et n. Dans les copies des matrices A1 et B, les éléments non écrits sont nuls.

Le dual du programme (PC) s'écrit :

$$(DC) \quad \begin{cases} v \in \mathbb{R}^{2 \cdot (n-2)} \\ u \in \mathbb{R}^{2 \cdot n} \\ v \cdot A1 - u \cdot B \leq 0 \\ u \cdot A2 \leq c \\ v \cdot 1 + u \cdot M = w(\max) \end{cases}$$

$$\begin{aligned} \text{Soit } D &= \{u/u \in \mathbb{R}^{2 \cdot n} ; u.A2 \leq c\} \\ &= \{u/u \in \mathbb{R}^{2 \cdot n} ; u_i + u_{n+j} \leq c_{ij} ; i = 1, \dots, n ; j = 1, \dots, n\} \end{aligned}$$

Pour $u \in D$, désignons par $w(u)$ la valeur de la solution optimale du programme linéaire (DC_u) défini par :

$$(DC_u) \begin{cases} v \in \mathbb{R}^{2 \cdot (n-2)} \\ v.A1 \leq u.B \\ v.1 + u.M = w(u) \text{ (max)} \end{cases}$$

Nous appelons problème (P') , le problème :

$$(P') = \max_{u \in D} w(u)$$

Proposition 1 : Le problème (P') est équivalent au programme linéaire (DC) en ce sens que si u^* est solution du problème (P') et v^* est solution optimale du programme (DC_{u^*}) , alors (u^*, v^*) est solution optimale du programme (DC) .

Démonstration : Evidente.

Cette proposition nous conduit à étudier le problème (P') . Pour commencer, voici quelques propriétés.

§2 - PROPRIETES DE LA FONCTION w

Signalons d'abord que dans la suite de l'exposé, sauf indication contraire, les indices en haut sont réservés pour numérotter les vecteurs, les indices en bas pour numérotter les scalaires .

2.1 - Problème d'affectation

Définition 1 : Le problème de transport particulier suivant [14] :

$$(PA) \begin{cases} y \geq 0 \\ A1.y = 1 \\ \sum_{i=2}^{n-1} \sum_{j=2}^{n-1} \ell_{ij} \cdot u_{ij} = z(\min) \end{cases}$$

où A_1 , y et $\mathbf{1}$ sont définis au paragraphe 1, et où $\{\ell_{ij}\}$ est un système de coûts réels, est appelé problème d'affectation de coût minimum.

Soit $G = (X_1, X_2 ; E)$ un graphe biparti complet et non orienté ; $X_1 = X_2 = \{2, 3, \dots, n-1\}$ sont les deux ensembles des sommets, $E = \{(i, j) / i \in X_1, j \in X_2\}$ l'ensemble des arêtes. Soit $\{\ell_{ij}\}$ un système de coûts réels sur les arêtes. On appelle couplage dans G un sous-ensemble d'arêtes de E tel que deux quelconques d'entre elles ne sont pas adjacentes [15]. La somme des coûts des éléments du couplage est appelée coût du couplage. Un couplage est parfait s'il sature tous les sommets de X_1 et de X_2 . Dans la suite, nous nous intéresserons aux couplages parfaits seulement et nous les appellerons tout simplement couplages.

Il est bien connu que toute solution de base y^0 du programme linéaire (PA) est entier ($y_{ij}^0 = 0, 1$) et que l'ensemble des $(n-2)$ arêtes $\{(i, j) / y_{ij}^0 = 1\}$ forme un couplage dans G . Le problème d'affectation (PA) est donc exactement le problème de couplage de coût minimum dans le graphe biparti G ; la matrice A_1 étant la matrice d'incidence aux arêtes. Le problème de couplage dans un graphe quelconque est polynomial et est facile à résoudre. Le problème d'affectation l'est davantage. Les deux lemmes suivants montrent que, certaines hypothèses étant faites sur le système coût $\{\ell_{ij}\}$, ce problème devient très facile.

Lemme 1 : Soient $(a_2, a_3, \dots, a_{n-1})$ et $(b_2, b_3, \dots, b_{n-1})$ deux systèmes de nombres réels en bijection avec les ensembles X_1 et X_2 respectivement : le sommet i de X_1 est en bijection avec a_i et le sommet j de X_2 est en bijection avec b_j . Supposons que $a_2 \geq a_3 \geq \dots \geq a_{n-1}$, $b_2 \leq b_3 \leq \dots \leq b_{n-1}$ et que le système coût $\{\ell_{ij}\}$ sur les arêtes vérifie la relation $\ell_{ij} = a_i \cdot b_j$ pour $i = 2, 3, \dots, n-1$; $j = 2, 3, \dots, n-1$; alors la solution du problème d'affectation (PA) est y^* telle que $y_{ii}^* = 1$ $i = 2, 3, \dots, n-1$ et $y_{ij}^* = 0$ ailleurs. (C'est à dire que le sommet i de X_1 est affecté au sommet i de X_2 pour $i = 2, 3, \dots, n-1$).

Démonstration : Supposons que l'affectation optimale n'est pas celle définie par le lemme 1, il existe donc, dans cette solution optimale :

$$2 \leq i < j \leq n-1, \quad a_i < a_j \quad (1)$$

$$2 \leq p < q \leq n-1, \quad b_p < b_q \quad (2)$$

tels que i soit affecté à p et j soit affecté à q .

Les relations (1) et (2) impliquent que $a_i (b_q - b_p) < a_j (b_q - b_p)$ ou que :

$$a_i \cdot b_q + a_j \cdot b_p < a_i \cdot b_p + a_j \cdot b_q \quad (3)$$

La relation (3) montre qu'on peut améliorer la solution obtenue en affectant au lieu de i à p et j à q , i à q et j à p . Ce qui contredit à l'optimalité.

Lemme 2 : Reprenons les mêmes notations que dans le lemme 1. Soit de plus J l'ensemble des sommets de X_1 qui sont défendus d'être affectés au sommet 2 de X_2 . Soit i_0 le plus petit des i qui n'appartiennent pas à J , alors la solution optimale s'obtient en affectant le sommet i_0 de X_1 au sommet 2 de X_2 , le sommet 2 de X_1 au sommet 3 de X_2 , ..., le sommet (i_0-1) de X_1 au sommet i_0 de X_2 , le sommet (i_0+1) de X_1 au sommet (i_0+1) de X_2 , ..., le sommet $(n-1)$ de X_1 au sommet $(n-1)$ de X_2 .

Démonstration : Par convention, si $J = X_1$, le coût de l'affectation sera pris égal à ∞ . Supposons que $J \neq X_1$. Montrons d'abord que, dans l'affectation optimale, i_0 ($= 4$ dans le cas de la figure 1) doit être affecté au sommet 2 de X_2 .

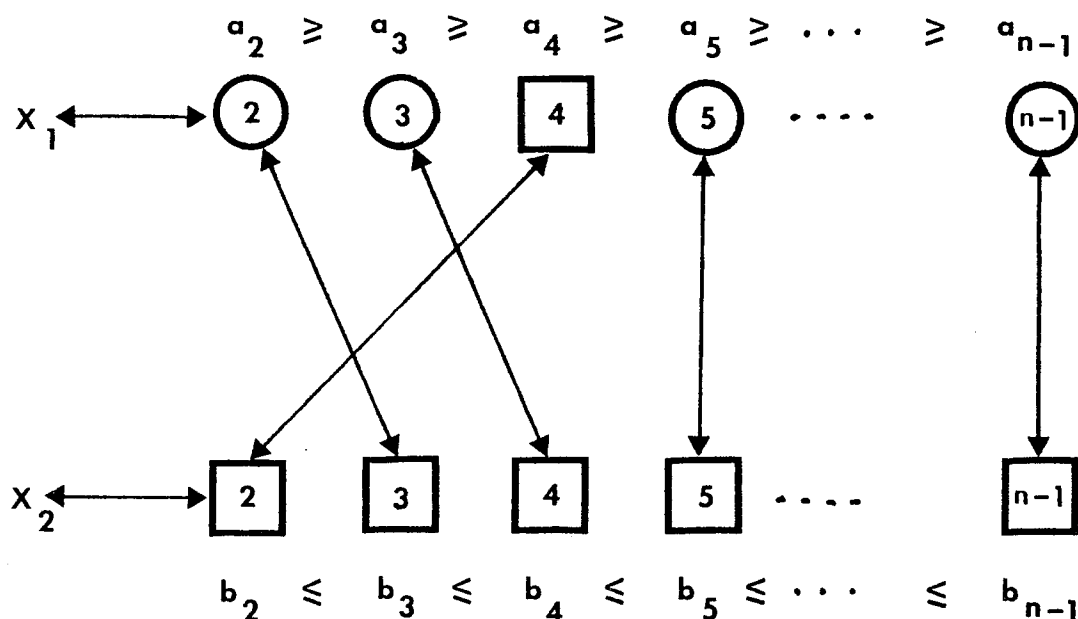


FIGURE 1

Supposons le contraire, il existe alors $j_o \in X_1$, $j_o \notin J$, $j_o > i_o$ tel que j_o soit affecté au sommet 2 de X_2 . Soit q_o le sommet de X_2 auquel est affecté i_o , alors les couples (i_o, j_o) et $(2, q_o)$ vérifient les relations de type (1) et (2) de la démonstration du lemme 1 et, de la même manière, on peut améliorer l'affectation. Ce qui contredit à l'optimalité. Donc dans la solution optimale, i_o de X_1 doit être affecté au sommet 2 de X_2 .

Pour achever la démonstration, il suffit d'appliquer le lemme 1 au graphe biparti complet $G' = (X'_1, X'_2 ; E')$ où $X'_1 = X_1 - \{i_o\}$ et $X'_2 = X_2 - \{2\}$. $\square$

2.2 - Nature de la fonction w

Théorème 1 : Soient $u = (u_1, u_2, \dots, u_n, u_{n+1}, u_{n+2}, \dots, u_{n+n}) \in D$ et $\sigma = (\sigma_2, \sigma_3, \dots, \sigma_{n-1}) \in \mathcal{S}_{[2, n-1]}$ tels que :

$$u_{\sigma_2} + u_{n+\sigma_2} \geq u_{\sigma_3} + u_{n+\sigma_3} \geq \dots \geq u_{\sigma_{n-1}} + u_{n+\sigma_{n-1}} \quad (4)$$

Soient $\sigma^{-1} = (\sigma_2^{-1}, \sigma_3^{-1}, \dots, \sigma_{n-1}^{-1}) \in \mathcal{S}_{[2, n-1]}$ la permutation inverse de σ et $\theta^\sigma \in R^{2 \cdot n}$ défini par :

$$\theta^\sigma = (m+1, m+\sigma_2^{-1}, m+\sigma_3^{-1}, \dots, m+\sigma_{n-1}^{-1}, 0, 0, m+\sigma_2^{-1}-1, m+\sigma_3^{-1}-1, \dots, m+\sigma_{n-1}^{-1}-1, m+n-1) \quad (5)$$

alors $w(u) = \langle u, \theta^\sigma \rangle$ ($\langle, \rangle$ désigne le produit scalaire).

Démonstration : Reprenons le programme linéaire (DC_u)

$$(DC_u) \begin{cases} v \in R^{2 \cdot (n-2)} \\ v.A1 \leq u.B \\ v.1 + u.M = w(u) \text{ (max)} \end{cases}$$

Son dual (PC_u) s'écrit :

$$(PC_u) \begin{cases} y \geq 0 \\ A1.y = 1 \\ u.B.y + u.M = z(u) \text{ (min)} \end{cases}$$

Ce n'est autre que le problème d'affectation décrit au 2-1 où le système coût $\{\ell_{ij}\} = \{(u.B)_{ij}\}$. La forme de la matrice B décrite au §1 implique que :

$$(u.B)_{ij} = (u_i + u_{n+i}) \cdot j - u_{n+i} \quad 2 \leq i, j \leq n-1 \quad (6)$$

En portant cette valeur de $(u.B)_{ij}$ dans la fonction objective de (PC_u) , on a :

$$\begin{aligned} u.B.y + u.M &= \sum_{i=2}^{n-1} \sum_{j=2}^{n-1} (u_i + u_{n+i}) \cdot j \cdot y_{ij} - \sum_{i=2}^{n-1} \sum_{j=2}^{n-1} u_{n+i} \cdot y_{ij} + u.M \\ &= \sum_{i=2}^{n-1} \sum_{j=2}^{n-1} (u_i + u_{n+i}) \cdot j \cdot y_{ij} - \sum_{i=2}^{n-1} u_{n+i} + u.M \end{aligned}$$

En posant $a_i = (u_i + u_{n+i})$, $b_j = j$ et en appliquant le lemme 1, à l'optimum de (PC_u) , on a :

$$w(u) = z(u) = \sum_{i=2}^{n-1} (u_{\sigma_i} + u_{n+\sigma_i}) \cdot i - \sum_{i=2}^n u_{n+i} + u.M$$

où $\sigma = (\sigma_2, \sigma_3, \dots, \sigma_{n-1}) \in \mathfrak{S}_{[2, n-1]}$ est telle que :

$$(u_{\sigma_2} + u_{n+\sigma_2}) \geq (u_{\sigma_3} + u_{n+\sigma_3}) \geq \dots \geq (u_{\sigma_{n-1}} + u_{n+\sigma_{n-1}})$$

Il suffit alors de nous rappeler la forme de M décrite au §1 pour achever la démonstration. $\square$

Notations 1 : Les deux notations suivantes seront adoptées dans toute la suite de l'exposé :

(i) Soit $\sigma \in \mathfrak{S}_{[2, n-1]}$, nous notons par θ^σ le $2.n$ -vecteur défini par la relation (5) du théorème 1.

(ii) Soit $u \in D$, nous désignons par $\mathfrak{S}(u)$ l'ensemble des permutations de $\mathfrak{S}_{[2, n-1]}$ qui vérifient la relation (4) du théorème 1. Il est clair que :

$$\mathfrak{S}(u) = \{\sigma / \sigma \in \mathfrak{S}_{[2, n-1]}, w(u) = \langle u, \theta^\sigma \rangle\}.$$

Corollaire du théorème 1 : Soient $(s_1, s_2, \dots, s_n)$ et $(t_1, t_2, \dots, t_n)$ deux systèmes de nombres réels tels que :

$$s_2 + t_2 \geq s_3 + t_3 \geq \dots \geq s_{n-1} + t_{n-1} \quad (7)$$

Supposons que dans le problème (P) de recherche d'une tournée optimale, le système coût c est défini par la relation :

$$c_{ij} = s_i + t_j \quad i = 1, 2, \dots, n \quad j = 1, 2, \dots, n \quad (8)$$

alors, la tournée $T^* = (1, 2, 3, \dots, n-1, n)$ est solution optimale du problème (P) et $u^* = (s_1, s_2, \dots, s_n, t_1, t_2, \dots, t_n)$ est solution optimale du problème (P').

Démonstration : Il est clair que $u^* \in D$. De plus, d'après le théorème 1, on a :

$$\begin{aligned} w(u^*) &= (m+1) \cdot s_1 + (m+2) \cdot s_2 + \dots + (m+n-1) \cdot s_{n-1} + \\ &\quad (m+1) \cdot t_2 + (m+2) \cdot t_3 + \dots + (m+n-2) \cdot t_{n-1} + (m+n-1) \cdot t_n \\ &= (m+1)(s_1 + t_2) + (m+2)(s_2 + t_3) + \dots + (m+n-1)(s_{n-1} + t_n) \\ &= (m+1) \cdot c_{12} + (m+2) \cdot c_{23} + \dots + (m+n-1) \cdot c_{(n-1)n} \end{aligned}$$

Soit $w(u^*) = c(T^*)$.

Or, on sait que $\forall u \in D, \forall T \in \mathcal{T}, w(u) \leq c(T)$.

Donc u^* et T^* sont respectivement solutions optimales de (P') et (P). $\square$

Définition 2 : On dit qu'un système coût c possède la propriété $c(F)$ s'il existe deux systèmes $(s_1, s_2, \dots, s_n)$ et $(t_1, t_2, \dots, t_n)$ de nombres réels tels que la relation (8) soit vérifiée.

Définition 3 : Soient c et c^1 deux systèmes de coût quelconques, on dit que le coût c^1 est inférieur à c et on écrit $c^1 \leq c$ si $\forall i \in \{1, 2, \dots, n\}, \forall j \in \{1, 2, \dots, n\}, c_{ij}^1 \leq c_{ij}$.

Remarque 1 : Soit un vecteur $u = (u_1, u_2, \dots, u_n, u_{n+1}, u_{n+2}, \dots, u_{n+n})$ et soit c^1 le coût possédant la propriété $c(F)$ défini par $c_{ij}^1 = u_i + u_{n+j}$ $i = 1, 2, \dots, n$ $j = 1, 2, \dots, n$. D'après le corollaire du théorème 1, le problème de recherche d'une tournée optimale par rapport au coût c^1 est très facile. Le fait que $u \in D$ implique que $c^1 \leq c$, donc que le coût de la tournée optimale par rapport au coût c^1 constitue une borne inférieure des coûts des tournées par rapport à c . C'est ainsi que le problème (P') est interprété comme la recherche d'un coût c^1 possédant la propriété $c(F)$, $c^1 \leq c$, par rapport auquel le coût de la tournée optimale constitue la meilleure borne des coûts des tournées par rapport à c .

Remarque 2 : Il est facile de voir que le domaine D est convexe [9]. De plus, d'après le théorème 1, pour $u \in D$, $w(u) = \min_{\sigma \in \mathcal{G}_{[2, n-1]}} \langle u, \theta^\sigma \rangle$.

Proposition 2 : La fonction $w : R^{2.n} \rightarrow \bar{R}$ définie par :

$$w(u) = \begin{cases} -\infty & \text{si } u \notin D \\ \min_{\sigma \in \mathcal{G}_{[2,n-1]}} \langle u, \theta^\sigma \rangle & \text{si } u \in D \end{cases}$$

est concave.

Démonstration : La fonction $w' : R^{2.n} \rightarrow \bar{R}$ définie par
 $w'(u) = \min_{\sigma \in \mathcal{G}_{[2,n-1]}} \langle u, \theta^\sigma \rangle$ est concave comme enveloppe inférieure d'une famille
 d'applications linéaires. Ce qui implique la concavité de la fonction w .

2.3 - Caractérisation du maximum de w

Définition 4 : Soient $k \in \mathbb{N}$ et $f : R^k \rightarrow \bar{R}$ une fonction concave quelconque. Soit $x_0 \in R^k$ tel que $f(x_0) \neq \pm \infty$. Un vecteur y_0 de R^k sera dit appartenir au sous-différentiel $\partial f(x_0)$ de la fonction f au point x_0 si la fonction $x \rightarrow f(x) - \langle x, y_0 \rangle$ atteint son maximum au point x_0 . Le vecteur y_0 s'appelle aussi sous-gradient de f au point x_0 .

Un des résultats les plus importants concernant les fonctions concaves ou convexes est le suivant :

Théorème 2 [9] : Une condition nécessaire et suffisante pour que la fonction concave f atteigne son maximum en x_0 est que $0 \in \partial f(x_0)$.

Ce théorème donne une caractérisation très importante du maximum d'une fonction concave et montre combien il est nécessaire, dans notre cas de connaître le sous-différentiel de la fonction w en un point u donné. Ceci fait l'objet de la proposition 3 suivante qui a été démontrée dans [11] dans un cas particulier, le cas où D est égal à $R^{2.n}$ tout entier. Mais avant tout, voici les deux notations que nous adopterons dans toute la suite.

Notations 2 :

(i) Soit $u \in D$, à chaque couple (i, j) tel que $u_i + u_{n+j} = c_{ij}$, nous associons un $2.n$ -vecteur $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_n, \gamma_{n+1}, \gamma_{n+2}, \dots, \gamma_{2n})$ défini par $\gamma_i = \gamma_{n+j} = 1$ et $\gamma_k = 0$ pour $k \neq i$ et $k \neq j$. Nous notons indifféremment pour $\Gamma(u)$ l'ensemble $\{\gamma^1, \gamma^2, \dots, \gamma^k\}$ de tous les vecteurs γ ainsi obtenus ou la matrice dont les colonnes sont les vecteurs $\gamma^1, \gamma^2, \dots, \gamma^p$.

(ii) Soit $u \in D$, nous notons indifféremment par $\theta(u) = \{\theta^1, \theta^2, \dots, \theta^q\}$ l'ensemble $\{\theta/w(u) = \langle u, \theta \rangle\}$ ou la matrice dont les colonnes sont les vecteurs $\theta^1, \theta^2, \dots, \theta^q$.

Proposition 3 : Soit $u^0 \in D$ et supposons que $\Gamma(u^0) = \{\gamma^1, \gamma^2, \dots, \gamma^p\}$ et $\theta(u^0) = \{\theta^1, \theta^2, \dots, \theta^q\}$, alors le sous-différentiel de w en u^0 est :

$$\partial w(u^0) = \{v/v = - \sum_{i=1}^p \alpha_i \cdot \gamma^i + \sum_{i=1}^q \lambda_i \cdot \theta^i ; \alpha_i \geq 0 \text{ pour } i = 1, 2, \dots, p ;$$

$$\lambda_i \geq 0 \text{ pour } i = 1, 2, \dots, q \text{ et } \sum_{i=1}^q \lambda_i = 1\}.$$

Démonstration : Désignons par θ la matrice dont les vecteurs colonnes sont les vecteurs θ^σ , $\sigma \in \tilde{S}_{[2, n-1]}$. D'après la définition 4, dire que $v^0 \in \partial w(u^0)$ équivaut à dire que la fonction $u \rightarrow w(u) - \langle u, v^0 \rangle$ atteint son maximum en u^0 ou encore que le programme linéaire

$$(PL) \begin{cases} u \cdot A2 \leq c \\ p - u \cdot \theta \leq 0 \\ p - \langle u, v^0 \rangle = w(\max), \end{cases}$$

où $p \in \mathbb{R}$, $A2$ connu au §1, θ d'ordre $2.n \times (n-2)!$, admet (ρ_0, u^0) comme solution optimale.

D'après le théorème de dualité, le programme linéaire dual

$$(DL) \begin{cases} \alpha = (\alpha_1, \alpha_2, \dots, \alpha_{n-2}) \geq 0 \\ \lambda = (\lambda_1, \lambda_2, \dots, \lambda_{(n-2)!}) \geq 0 \\ \begin{pmatrix} 0 & 0 & \dots & 1 & 1 & \dots & 1 \\ \hline & A2 & & & -\theta & \end{pmatrix} \cdot \begin{pmatrix} \alpha \\ \lambda \end{pmatrix} = \begin{pmatrix} 1 \\ -v^0 \end{pmatrix} \\ c. = z(\min) \end{cases}$$

admet une solution optimale (α, λ) . Le théorème des écarts complémentaires affirme que :

(i) Pour les contraintes $u_i^0 + u_{n+j}^0 < c_{ij}$, les composantes correspondantes de α sont nulles.

(ii) Pour les contraintes $\rho < \langle u, \theta \rangle$, les composantes correspondantes de λ sont nulles.

Il existe donc deux systèmes $(\alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_p})$ et $(\lambda_{j_1}, \lambda_{j_2}, \dots, \lambda_{j_q})$ non négatifs tels que $\lambda_{j_1} + \lambda_{j_2} + \dots + \lambda_{j_q} = 1$ et :

$$v^0 = - \sum_{k=1}^q \alpha_{i_k} \cdot \gamma_{i_k}^k + \sum_{k=1}^q \lambda_{j_k} \cdot \theta_{j_k}^k \quad (9)$$

Donc $v^0 \in \partial w(u^0)$ implique que v^0 vérifie la relation (9).

La réciproque est immédiate.

Notation 3 : Soit $u \in D$, nous notons par $A(u)$ la matrice d'ordre $(2.n+1) \times (p+q)$ définie par :

$$A(u) = \left(\begin{array}{ccc|ccc} 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 \\ \hline & & & & & & & \\ \Gamma(u) & & & & & & & -\Theta(u) \end{array} \right)$$

où $\Gamma(u)$ et $\Theta(u)$ sont définis dans la notation 2.

Théorème 3 : Soit $u \in D$. Les deux propositions suivantes sont équivalentes :

- (i) u n'est pas solution optimale du problème (P').
- (ii) Il existe $(\rho, d) \in \mathbb{R}^{2.n+1}$ ($\rho \in \mathbb{R}$, $d \in \mathbb{R}^{2.n}$) tel que $\rho > 0$ et $(\rho, d) \cdot A(u) \leq 0$.

Démonstration : Soit $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_p) \in \mathbb{R}^p$ et $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_q) \in \mathbb{R}^q$.

Considérons le système :

$$\left(\begin{array}{l} (\alpha, \lambda) \geq 0 \\ A(u) \cdot \begin{pmatrix} \alpha \\ \lambda \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \end{array} \right) \begin{array}{c} \updownarrow \\ 2n+1 \end{array}$$

D'après le théorème des alternatives [14] [18], l'une et l'une seulement des deux propositions suivantes est vraie :

$$(a) \exists (\alpha, \lambda) \geq 0 \text{ tel que } A(u) \cdot \begin{pmatrix} \alpha \\ \lambda \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$(b) \exists (\rho, d) \in \mathbb{R}^{2.n+1} \text{ tel que } (\rho, d) \cdot A(u) \leq 0 \text{ et } (\rho, d) \cdot \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} > 0$$

Le cas (a) équivaut aussi à dire que $0 \in \partial w(u)$.

Si u n'est pas solution optimale du problème (P'), alors $0 \notin \partial w(u)$ et par conséquent, on est dans le cas (b) ; ce qui démontre que (i) $\Rightarrow$ (ii).

Réciproquement, si on est dans le cas (ii) ou (b), alors on n'est pas dans le cas (a) et $0 \notin \partial w(u)$. $\square$

Ce théorème, dont la démonstration est très simple, donne une caractérisation particulièrement commode de l'optimum de la fonction concave w .

De plus, cette démonstration fournit un principe très important sur lequel se basent toutes les méthodes de résolution du problème (P') que nous développerons dans le chapitre suivant.

CHAPITRE III

METHODES DE RESOLUTION
DU
PROBLEME (P')

§1 - METHODE ASCENDANTE

Dans cette partie, en plus des notations des chapitres précédents que nous reprenons dans toute la suite, nous adoptons :

Notation 1 :

(i) Nous désignons par θ la matrice dont l'ensemble des colonnes est $\{\theta^\sigma / \sigma \in \mathcal{G}_{[2, n-1]}\}$

(ii) Nous désignons par b le $(2n+1)$ -vecteur défini par $b_1 = 1$ et $b_i = 0$ pour $i = 2, 3, \dots, 2n+1$.

Soit $u \in D$, le théorème 3 du chapitre II affirme que, si u n'est pas solution optimale de (P') , alors il existe $(\rho, d) \in R^{2n+1}$ ($\rho \in R, d \in R^{2.n}$) tel que :

$$\rho > 0 \quad (1)$$

$$(\rho, d) \cdot A(u) \leq 0 \quad (2)$$

La relation (2) s'exprime aussi en disant que

$$\forall \gamma \in \Gamma(u), \quad \langle d, \gamma \rangle \leq 0 \quad (3)$$

$$\forall \theta \in \Theta(u), \quad \langle d, \theta \rangle \geq \rho$$

ou encore, puisque que $\rho > 0$,

$$\forall \theta \in \Theta(u), \quad \langle d, \theta \rangle > 0 \quad (4)$$

Pour un réel positif t suffisamment petit, le point $u+t.d$ possède les propriétés suivantes :

(i) $u+t.d \in D$.

En effet, d'après (3), pour le couple (i, j) tel que $u_i + u_{n+j} = c_{ij}$, on a :

$$u_i + u_{n+j} + t.d_i + t.d_{n+j} = u_i + u_{n+j} + t.\langle d, \gamma \rangle \quad \text{où} \quad \langle d, \gamma \rangle \leq 0$$

(ii) $w(u+t.d) > w(u)$

En effet,

$$\begin{aligned} w(u+t.d) &= \min_{\theta \in \Theta(u)} (\langle u, \theta \rangle + t.\langle d, \theta \rangle) \\ &= w(u) + t. \min_{\theta \in \Theta(u)} \langle d, \theta \rangle \end{aligned} \quad (5)$$

$> w(u)$, d'après la relation (4).

Le vecteur d ainsi obtenu s'appelle une direction de montée de la fonction w au point u . Un résultat analogue est énoncé comme théorème 3 dans [5] dans un cas particulier, le cas où le domaine D est égal à $\mathbb{R}^{2.n}$.

Soit $u \in D$ et supposons que u n'est pas une solution du problème (P'). Soit d une direction de montée de w en u . Posons :

$$\Delta(u,d) = \min_{\theta \in \Theta(u)} \langle d, \theta \rangle$$

$$\Theta(u,d) = \{ \theta / \theta \in \Theta(u) \text{ et } \langle d, \theta \rangle = \Delta(u,d) \}$$

alors, la relation (5) nous permet d'énoncer :

Proposition 1 : Pour un réel positif t suffisamment petit, on a :

$$\Theta(u+t.d) = \Theta(u,d) \quad \text{et} \quad w(u+t.d) = w(u) + t.\Delta(u,d)$$

L'algorithme que nous allons proposer consiste à, partant d'un point $u \in D$, déterminer une direction de montée de w en u (ou prouver que u est une solution optimale de (P') si d n'existe pas), déterminer le plus grand t tel que $\Theta(u+t.d) = \Theta(u,d)$ et $u+t.d \in D$, poser $u' = u+t.d$ et recommencer la même opération si u' n'est toujours pas une solution optimale de (P').

Remarque 1 :

Considérons le programme linéaire maître [3] :

$$(PLM) \quad \begin{cases} x \geq 0, \phi \geq 0 & (\phi \in \mathbb{R}^{(n-2)!}) \\ \sum_{i=1}^{(n-2)!} \phi_i = 1 \\ A2.x - \Theta.\phi = 0 & (\text{notation 1}) \\ c.x & = z(\min) \end{cases}$$

$$(\Theta.\phi = \sum_{i=1}^{(n-2)!} \phi_i . \theta^i \text{ où l'indice } i \text{ parcourt l'ensemble des permutations } \mathfrak{S}_{[2,n-1]})$$

Son dual est

$$(DLM) \quad \begin{cases} u.A2 & \leq c \\ \rho - \langle u, \theta \rangle \leq 0 & , \quad \theta \in \Theta \\ \rho & = w(\max) \end{cases}$$

L'application de l'algorithme dual révisé du simplexe [3], [14], [18] sur le programme (PLM) revient à déterminer d'abord une base duale réalisable de (PLM) qui correspond à une variable duale (ρ, u) telle que $u \in D$ et $\rho - \langle u, \theta \rangle \leq 0$ pour tout $\theta \in \Theta$, ensuite à pivoter sur cette base pour obtenir une autre base toujours duale réalisable de (PLM) et ainsi de suite pour aboutir finalement à une base qui est à la fois duale et primale réalisable. Les développements plus haut montre que la procédure que nous allons proposer n'est autre que l'algorithme dual révisé du simplexe appliqué sur le programme-maitre (PLM) avec une technique de génération de colonne, du type Benders [3].

A partir de cette remarque, nous nous contentons de décrire la procédure en nous passant de certaines justifications, les lecteurs qui s'y intéressent peuvent les trouver dans [3].

1.1 - Description de l'algorithme

Initialisations

Elles consistent à déterminer un meilleur point u de départ appartenant à D . Nous en donnons deux, la première est nettement moins bonne que la seconde mais elle permet de mieux comparer les méthodes de résolution du problème (P') entre elles.

Initialisation (1)

Posons :

$$u_i^0 = \min_{1 \leq j \leq n} c_{ij} \quad , \quad i = 1, 2, \dots, n$$

$$u_{n+i}^0 = 0 \quad , \quad i = 1, 2, \dots, n$$

Initialisation (2)

Posons :

$$u_i^0 = \min_j c_{ij} \quad , \quad i = 1, 2, \dots, n$$

$$u_{n+i}^0 = \min_{1 \leq j \leq n} \{c_{ji} - u_j\} \quad , \quad i = 1, 2, \dots, n$$

Il est facile de voir que les deux points u^0 ainsi construits appartiennent à D .

Construction d'une direction de montée

Soit $u \in D$, considérons la matrice $A(u)$ associée à u . Nous supposons que l'hypothèse de non dégénérescence est vérifiée c'est à dire que le nombre de colonnes de la matrice $\Gamma(u)$ est inférieur ou égal à $2n$. De plus, nous pouvons supposer que le nombre de colonnes de $A(u)$ est inférieur ou égal à $2n+1$ (sinon, nous remplaçons u_i par u_i/t où t est un réel convenablement choisi). Reprenons donc le système :

$$\begin{cases} (\alpha, \lambda) \geq 0 \\ A(u) \cdot \begin{bmatrix} \alpha \\ \lambda \end{bmatrix} = b \end{cases} \quad (\text{notation 1})$$

où $A(u)$ est d'ordre $(2n+1) \times k$ avec $k \leq 2n+1$. Soit A la matrice d'ordre $(2n+1) \times (2n+1)$ telle que $A \cdot A(u) = \bar{A}(u)$. Par une permutation de lignes et de colonnes de $A(u)$, on peut supposer que $\bar{A}(u)$ possède la forme suivante :

$$\bar{A}(u) = \begin{pmatrix} 1 & & & & & \\ & 1 & & & 0 & \\ & & \ddots & & & \\ & 0 & & \ddots & & \\ & & & & 1 & \\ & & & & & 0 \end{pmatrix} \quad \begin{matrix} \updownarrow k \\ \updownarrow 2n+1 \end{matrix}$$

Définition 1 : Nous disons que A est la matrice inverse de la base par rapport au point u .

Soit $\bar{b} = A \cdot b =$ première colonne de A .

1er cas : $\exists r, 1 \leq r \leq k$ tel que $\bar{b}_r < 0$, alors le vecteur $(\rho, d) = (0, \dots, 0, -1, 0, \dots, 0)$. A , où la composante non nulle égale à -1 est de rang r , vérifie les relations :

$$\begin{aligned} (\rho, d) \cdot A(u) &= (0, \dots, 0, -1, 0, \dots, 0) \cdot A \cdot A(u) \\ &= (0, \dots, 0, -1, 0, \dots, 0) \cdot \bar{A}(u) \leq 0 \end{aligned} \quad (6)$$

$$(\rho, d) \cdot b = (0, \dots, 0, -1, 0, \dots, 0) \cdot \bar{b} > 0 \quad (7)$$

Le vecteur d ainsi déterminé est une direction de montée de w en u .

2ème cas : $\forall r \in \{1, 2, \dots, k\}, \bar{b}_r \geq 0$ et il existe $r, k+1 \leq r \leq 2n+1$ tel que $\bar{b}_r > 0$ (resp. $\bar{b}_r < 0$), alors on vérifie facilement que le vecteur

$(\rho, d) = (0, \dots, 0, +1, 0, \dots, 0) \cdot A$ (resp. $(0, \dots, 0, -1, 0, \dots, 0) \cdot A$), où la composante non nulle égale à +1 (resp. -1) est de rang r , vérifie les relations (6) et (7), et d sera toujours une direction de montée de w en u .

3ème cas : $\forall r \in \{1, 2, \dots, k\}, \bar{b}_r \geq 0$ et $\forall r \in \{k+1, k+2, \dots, 2n+1\}, \bar{b}_r = 0$. Dans ce cas, le k -vecteur $\bar{b}' = (\bar{b}_1, \bar{b}_2, \dots, \bar{b}_k)$ vérifie

$$\bar{b}' \geq 0 \quad (8)$$

$$\begin{aligned} A(u) \cdot \bar{b}' &= A^{-1} \cdot A \cdot A(u) \cdot \bar{b}' \\ &= A^{-1} \cdot \bar{A}(u) \cdot \bar{b}' \\ &= A^{-1} \cdot \bar{b} \\ &= b \end{aligned} \quad (9)$$

D'après le théorème 3 du chapitre II, (8) et (9) expriment bien que u est une solution optimale du problème (P') .

Construction du plus grand pas t

Soit $I_1 = \{(i, j) / u_i + u_{n+j} < c_{ij} \text{ et } d_i + d_{n+j} > 0\}$.

Posons $t' = \min_{(i, j) \in I_1} (c_{ij} - u_i - u_{n+j}) / (d_i + d_{n+j})$

D'autre part, pour simplifier l'exposé, supposons que le point u possède la propriété suivante :

$$u_2 + u_{n+2} = \dots = u_{i_1} + u_{n+i_1} > \dots > u_{i_k} + u_{n+i_k} = \dots = u_{n-1} + u_{2n-1}$$

De même, supposons que la direction de montée d possède la propriété suivante :

$$\begin{aligned} d_2 + d_{n+2} &\geq \dots \geq d_{i_1} + d_{n+i_1} \\ \dots &\dots \dots \\ \dots &\dots \dots \\ d_{i_k} + d_{n+i_k} &\geq \dots \geq d_{n-1} + d_{2n-1} \end{aligned}$$

Soit $I_2 = \{j / j = 1, 2, \dots, k, d_{i_j+1} + d_{n+i_j+1} > d_{i_j} + d_{n+i_j}\}$.

Posons $t'' = \min_{j \in I_2} (u_{i_j} + u_{n+i_j} - u_{i_j+1} - u_{n+i_j+1}) / (d_{i_j+1} + d_{n+i_j+1} - d_{i_j} - d_{n+i_j})$.

Alors :

$$t = \min\{t', t''\}$$

et $u' = u + t \cdot d$.

Remarque 2 :

Si u n'est pas solution optimale du problème (P') , d'après la construction du vecteur d , on se trouve soit dans le 1er cas, soit dans le 2ème cas. Si l'on est dans le 1er cas on doit quitter la $r^{\text{ème}}$ colonne de $A(u)$ qui correspond à une colonne ℓ^r de la matrice des contraintes du programme maître (PLM). Si l'on est dans le 2ème cas, aucune colonne de $A(u)$ n'est abandonnée, c'est une colonne correspondant à une variable artificielle de (PLM) qui est abandonnée.

Remarque 3 :

Supposons toujours que u n'est pas une solution optimale du problème (P') et on a donc $u' = u + t.d$. Si $t = t'$, en suivant la direction d , on "se butte" sur un hyperplan vertical et c'est une colonne de la matrice A_2 qui entre dans la base correspondant à u' . Si $t = t''$, on "se butte" sur un hyperplan défini par une colonne de $\Theta(u')$ et c'est une colonne de la matrice Θ qui entre dans la base correspondant à u' . Dans tous les cas, c'est une colonne $\bar{\ell}^S$ de la matrice des contraintes de (PLM) qui entre dans la base. C'est à ce niveau qu'intervient la technique de génération de colonne qui a été résolue par le lemme 1 du chapitre II et qui explique le calcul de t' et de t'' .

Remarque 4 :

Supposons que, en passant du point u au point u' , il existe une colonne ℓ^r qui quitte la base relative à u et une colonne $\bar{\ell}^S$ qui entre dans la base relative à u' . Il est clair qu'à l'étape actuelle de l'algorithme, il faut considérer, non pas $\bar{\ell}^S$, mais $\ell^S = A.\bar{\ell}^S$.

$$\bar{A}(u) = \begin{pmatrix} 1 & & 0 & & & \\ & 1 & & & & \\ & & \ddots & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 1 \\ & & & 0 & & & \\ & & & & \ddots & & \\ & & & & & & 1 \\ & & & & & & & \ddots & \\ & & & & & & & & 0 \end{pmatrix}$$

$\xleftarrow{r^e \text{ composante}}$

$$\ell^S = \begin{pmatrix} \ell_1^S \\ \ell_2^S \\ \vdots \\ \ell_r^S \\ \vdots \\ \ell_{2n+1}^S \end{pmatrix}$$

$\uparrow$
 ℓ^r

(Les éléments non écrits de la matrice $\bar{A}(u)$ sont nuls).

Pour avoir la matrice $\bar{A}(u')$, il suffit de remplacer la colonne ℓ^r de $\bar{A}(u)$ par ℓ^s , et de multiplier à droite la matrice ainsi obtenue par la matrice de pivotage [15] :

$$D_{rs} = \begin{pmatrix} 1 & -\frac{\ell_1^s}{\ell_r^s} & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & \frac{1}{\ell_r^s} & \\ & & & & \ddots \\ & & & & & \ell_{2n+1}^s \\ & & & -\frac{\ell_{2n+1}^s}{\ell_r^s} & & 1 \end{pmatrix}$$

La nouvelle matrice inverse de base par rapport à u' sera $D_{rs}.A$. Dans le cas où aucune colonne ne sort de l'ancienne base, on choisit l'indice r arbitrairement sur une composante ℓ_r^s non nulle de ℓ^s avec $r \neq 1, 2, \dots, k$.

1.2 - Algorithme

A chaque étape i de l'algorithme, la variable courante sera notée par u^i , le nombre de colonnes de la matrice $A(u^i)$ par k_i , la matrice inverse de la base par A^i , le vecteur $A^i.b$ par $\bar{b}^i$, le pas t par t_i , ρ par ρ_i , d par d^i , $\bar{\ell}^s$ par $\bar{\ell}_i^s$, ℓ^s par ℓ_i^s , ℓ^r par ℓ_i^r et D_{rs} par D_{rs}^i .

Pas (0) - Poser $i := 0$; définir u^0 par l'initialisation (1) ; calculer la matrice $A(u^0)$ d'ordre $(2n+1) \times k_0$; calculer la matrice inverse de base A^0 ainsi que $\bar{b}^0$.

Pas (1) - Si $\forall r \in \{1, 2, \dots, k_i\}, \bar{b}_r^i \geq 0$ et $\forall r \in \{k_i+1, k_i+2, \dots, 2n+1\}, \bar{b}_r^i = 0$, terminer : la solution u^i est optimale sinon :

1er cas : $\exists r \in \{1, 2, \dots, k_i\}, \bar{b}_r^i < 0$, aller en (2)

2ème cas : $\forall r \in \{1, 2, \dots, k_i\}, \bar{b}_r^i \geq 0$. $\exists r \in \{k_i+1, k_i+2, \dots, 2n+1\}, \bar{b}_r^i > 0$ (resp. $\bar{b}_r^i < 0$), aller en (3) :

Pas (2) : Poser $(\rho_i, d^i) := (0, \dots, 0, -1, 0, \dots, 0).A^i$ où la composante -1 est de rang r ; calculer le pas t_i ; faire $u^{i+1} := u^i + t_i.d^i$. Calculer la colonne à faire entrer dans la base $\bar{\ell}_i^s$; faire $\ell_i^s := A^i.\bar{\ell}_i^s$; calculer

la matrice de pivotage D_{rs}^i ; faire $A^{i+1} := D_{rs}^i \cdot A^i$, $k_{i+1} := k_i$,
 $\bar{b}^{i+1} :=$ première colonne de A^{i+1} , $i := i+1$ et aller en (1).

Pas (3) - Poser $(\rho_i, d^i) := (0, \dots, 0, +1, 0, \dots, 0) \cdot A^i$ (resp. $(\rho_i, d^i) := (0, \dots, 0, -1, 0, \dots, 0) \cdot A^i$) où la composante $+1$ (resp. -1) est de rang r ; calculer le pas t_i ; faire $u^{i+1} := u^i + t_i \cdot d^i$; calculer la colonne à faire entrer dans la base $\bar{\ell}_i^s$; faire $A^{i+1} := D_{rs}^i \cdot A^i$, $k_{i+1} := k_i + 1$, $\bar{b}^{i+1} :=$ première colonne de A^{i+1} , $i := i+1$ et aller en (1).

1.3 - Efficacité pratique

L'inconvénient de la méthode ascendante provient d'une part de la dégénérescence qui est liée à la nature du problème et d'autre part des erreurs de précision qui conduisent souvent la variable courante en dehors du domaine .

En effet, dans le programme (PLM), les coûts correspondant aux variables ϕ sont nuls ; ce qui implique que le polyèdre du programme (DLM) est très dégénéré. Prenons par exemple un système de coûts $c \geq 0$. Le point (ρ, u) où $\rho = 0$ et $u = 0$ appartient à au moins $(n-2)!$ hyperplans : ce sont les hyperplans qui correspondent aux colonnes de la matrice θ .

Il nous semble que, en pratique, ce phénomène de dégénérescence ne constitue pas le principal inconvénient de la méthode, ce sont plutôt les erreurs de précision qui posent de sérieux problèmes. En effet, pour les programmes de tailles assez grandes ($n \geq 10$), les erreurs de précisions entraînées par l'inversion d'une matrice d'ordre $(2n+1) \times (2n+1)$ accentuent nettement leurs effets. Etant donné un point $u \in D$, soit d la direction de montée calculée en u . Théoriquement, pour t suffisamment petit, $u + t \cdot d \in D$. En pratique, on avait souvent $u + t \cdot d \notin D$ (figure 1).

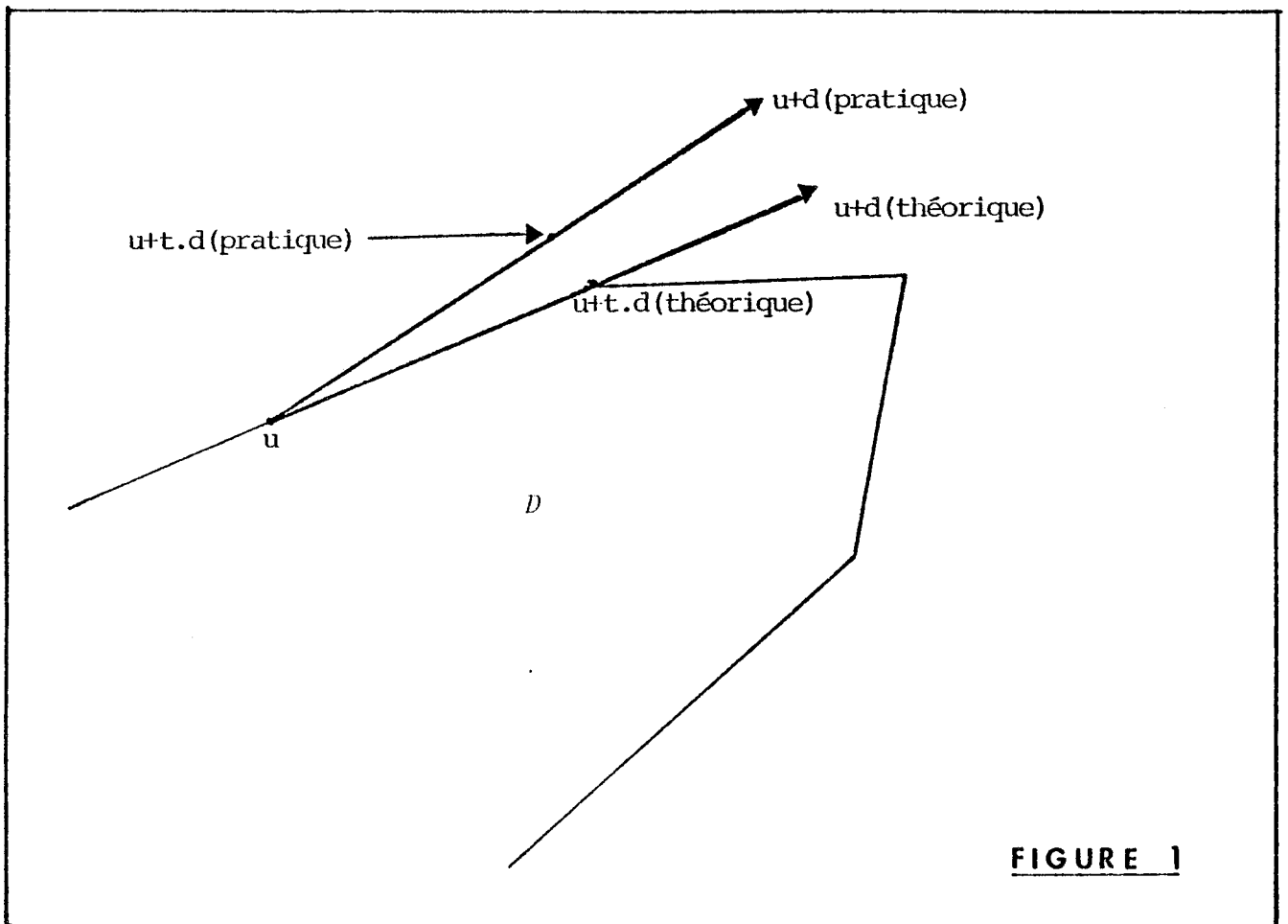


FIGURE 1

§2 - METHODE DE PENALITE

Les difficultés rencontrées dans la méthode ascendante, les erreurs de précision qui conduisent souvent la variable courante u en dehors du domaine D nous suggère d'envisager la suppression de ce domaine. Cette suppression nécessite une légère modification de la formulation proposée.

2.1 - Modification de la formulation

Reprenons le programme linéaire continu (PC)

$$(PC) \quad \left\{ \begin{array}{l} y \geq 0 \\ x \geq 0 \\ A1.y \quad \quad = 1 \\ -B.y + A2.x = M \\ \quad \quad \quad c.x = z(\min) \end{array} \right.$$

Nous avons vu que, une tournée étant donnée, les variables de positions y et de trafics x qui lui correspondent constituent une solution réalisable de (PC). D'autre part, il est clair que, pour cette même tournée, la somme totale des trafics sur les arcs est :

$$\sum_{i,j} x_{ij} = (2m+n)(n-1)/2 \quad (10)$$

que nous abrégons par

$$A3.x = a \quad (11)$$

Donc l'introduction de cette contrainte (10) ou (11) dans (PC) donne un nouveau programme (PCM) :

$$(PCM) \quad \left(\begin{array}{ll} y \geq 0 \\ x \geq 0 \\ A1.y & = 1 \\ -B.y + A2.x = M \\ & A3.x = a \\ & c.x = z(\min) \end{array} \right.$$

dont la valeur de la solution optimale reste toujours une borne inférieure des coûts des tournées. On peut penser que l'addition de cette contrainte supplémentaire à (PC) relève un peu la valeur de sa solution optimale, c'est à dire que le programme (PCM) donne une borne plus serrée des coûts des tournées. La proposition suivante affirme le contraire.

Proposition 2 : La contrainte (11) est redondante.

Démonstration : En effet, parmi les contraintes de (PC), on trouve (voir §3.1 du chapitre I) :

$$\begin{aligned} \sum_{j=1}^n x_{1j} &= m+1 \\ \sum_{j=1}^n x_{nj} &= 0 \\ \sum_{j=1}^n x_{ij} &= \sum_{j=2}^{n-1} j \cdot y_{ij} + m \quad i = 2, 3, \dots, n-1. \end{aligned}$$

En faisant la somme de toutes ces contraintes, on obtient :

$$\begin{aligned} \sum_{i,j} x_{ij} &= m+1 + \sum_{j=2}^{n-1} j \cdot \sum_{i=2}^{n-1} y_{ij} + (n-2) \cdot m \\ &= m+1 + \sum_{j=2}^{n-1} j + (n-2) \cdot m \\ &= (2m+n)(n-1)/2. \quad \square \end{aligned}$$

Si l'introduction de cette contrainte supplémentaire ne permet pas d'améliorer la borne cherchée, elle permet de supprimer le domaine D et d'appliquer une méthode proposée dans [6] pour résoudre le programme (PCM) ainsi obtenu. En effet, le dual de (PCM) est :

$$(DCM) \quad \left\{ \begin{array}{l} v \in R^{2 \cdot (n-2)} \\ u \in R^{2 \cdot n} \\ t \in R \\ v.A1 - u.B \leq 0 \\ u.A2 + t.A3 \leq c \\ v.1 + u.M + t.a = w(\max) \end{array} \right.$$

Pour chaque u fixé, on obtient un programme

$$(DCM_u) \quad \left\{ \begin{array}{l} v \in R^{2 \cdot (n-2)} \\ t \in R \\ v.A1 \leq u.B \\ t.A3 \leq c - u.A2 \\ v.1 + t.a + u.M = w(u) (\max) \end{array} \right.$$

Son dual s'écrit :

$$(PCM_u) \quad \left\{ \begin{array}{l} y \geq 0 \\ x \geq 0 \\ A1.y = 1 \\ A3.x = a \\ u.B.y + (c - u.A2).x + u.M = z(u) (\min) \end{array} \right.$$

Soit $w_1 : R^{2 \cdot n} \rightarrow R$ définie par $w_1(u) =$ valeur de la solution optimale de (PCM_u) . Il est clair que, si u^* est solution du problème :

$$\max_{u \in R^{2 \cdot n}} w_1(u)$$

et si v^* et t^* forment une solution optimale de (DCM_{u^*}) , alors (v^*, u^*, t^*) est une solution optimale de (DCM).

2.2 - Propriétés de la fonction w_1

Proposition 3 : $w_1(u) = \langle u, \theta^\sigma \rangle + \min_{i,j} \{c_{ij} - u_i - u_{n+j}\} \cdot (2m+n)(n-1)/2$ où $\sigma \in \mathcal{O}(u)$.

Démonstration : Il est évident que, si y^* est solution optimale de

$$(P1) \quad \begin{cases} y \geq 0 \\ A1.y = 1 \\ u.B.y + u.M = z'(u) \text{ (min)} \end{cases}$$

et x^* est solution de

$$(P2) \quad \begin{cases} x \geq 0 \\ A3.x = a \\ (c-u.A2).x = z''(u) \text{ (min)} \end{cases}$$

Alors (y^*, x^*) est une solution optimale de (PCM_u) . D'après le théorème 1 du chapitre I, à l'optimum de (P1), on a :

$$z'(u) = \langle u, \theta^\sigma \rangle \text{ avec } \sigma \in \tilde{\mathcal{G}}(u).$$

D'autre part, puisque (P2) ne possède qu'une seule contrainte :

$$\sum_{i,j} x_{ij} = (2m+n)(n-1)/2,$$

il est évident que, à l'optimum de (P2), on a :

$$z''(u) = \min_{i,j} \{c_{ij} - u_i - u_{n+j}\} \cdot (2m+n)(n-1)/2. \quad \square$$

Proposition 4 : La fonction w_1 est concave.

Démonstration : Comme w_1 est la somme de deux fonctions z' et z'' , pour prouver que w_1 est concave, il suffit de prouver que z' et z'' sont concaves.

z' est concave d'après la proposition 2 du chapitre II.

z'' est concave comme enveloppe inférieure d'une famille de fonctions affines. $\square$

Remarque 5 : Si l'on désigne par Q_1 l'ensemble des points extrêmes du polyèdre convexe associé au programme (P1), par Q_2 ceux du polyèdre convexe associé au programme (P2) et par $\{(y^1, x^1), (y^2, x^2), \dots, (y^q, x^q)\}$ le produit cartésien $Q_1 \times Q_2$, il est facile de voir que :

$$w_1(u) = \min_{1 \leq k \leq q} \{c^k + \langle u, v^k \rangle\} \quad (12)$$

$$\text{où } c^k = c.x^k \quad (13)$$

$$v^k = B.y^k + M - A2.x^k \quad (14)$$

La proposition suivante n'est qu'un cas particulier de la proposition 3 du chapitre II, cas où le domaine D est égal à $R^{2.n}$ tout entier.

Proposition 5 : Soient $u \in R^{2 \cdot n}$ et $K(u) = \{k/1 \leq k \leq q, w_1(u) = c^k + \langle u, v^k \rangle\}$, alors le sous-différentiel de w_1 en u est l'enveloppe convexe de l'ensemble $\{v^k/k \in K(u)\}$.

Cette proposition montre combien il est nécessaire de connaître explicitement la forme du vecteur v^k , pour $k \in K(u)$. Ceci fait l'objet du résultat suivant qui se déduit facilement de la remarque 5.

Proposition 6 : Soit $u \in R^{2 \cdot n}$. Soient $\sigma \in \mathcal{G}(u)$ et (i_0, j_0) tel que $c_{i_0} - u_{i_0} - u_{n+j_0} = \min_{i,j} \{c_{ij} - u_i - u_{n+j}\}$, alors, pour $k \in K(u)$,

$$v^k = \theta^0 + (0, \dots, 0, -a, 0, \dots, 0, -a, 0, \dots, 0) \quad (15)$$

où les deux composantes non nulles (égales à $-a$) sont de rang i_0 et $(n+j_0)$ respectivement.

2.3 - Description du premier algorithme

La procédure que nous proposons ici est due à Held et Karp [6]. Pour la décrire, nous avons besoin de la notation suivante :

Notation : Soit $u \in R^{2 \cdot n}$, nous désignons par $K(u)$ l'ensemble $\{v^k/1 \leq k \leq q, w_1(u) = c^k + \langle u, v^k \rangle\}$ et par $k(u)$ un élément quelconque de $K(u)$.

Cette notation étant faite, la procédure consiste alors à calculer une suite $\{u^i\}$ par la formule de récurrence suivante :

$$u^{i+1} = u^i + \bar{t} \cdot v^{k(u^i)} \quad (16)$$

où $\bar{t}$ est un scalaire égal à une constante.

Les résultats suivants (proposition 7, 8 et théorème 1) sont dans [6]. Certains d'entre eux sont des cas particuliers de ce qu'on peut trouver dans [12] et [13] concernant la méthode de relaxation pour résoudre un système d'inéquations. Ils établissent, en quelque sorte, la convergence de la procédure.

2.4 - Convergence de la première procédure

Proposition 7 : Soient $\bar{u}$ et u deux points de $R^{2 \cdot n}$ tels que $w_1(\bar{u}) \geq w_1(u)$, alors $(\bar{u}-u) \cdot v^{k(u)} \geq w_1(\bar{u}) - w_1(u) \geq 0$.

Donc l'hyperplan, passant par u et admettant $v^{k(u)}$ comme normale, détermine un demi-espace fermé contenant tous les points $\bar{u}$ tels que $w_1(\bar{u}) \geq w_1(u)$.

Ce demi-espace contient en particulier u^* , solution optimale de la fonction concave w_1 . La procédure proposée consiste justement à, partant d'un point u , faire un pas dans ce demi-espace en suivant la direction de la normale $v^k(u)$.

La proposition suivante montre que, pour t convenablement choisi, le point $u+t.v^k(u)$ est plus "proche" de $\bar{u}$ que u .

Proposition 8 : Si $0 < t < 2(w_1(\bar{u}) - w_1(u)) / \|v^k(u)\|^2$, alors $\|\bar{u} - (u+t.v^k(u))\| \leq \|\bar{u} - u\|$. ($\|\cdot\|$ désigne la norme euclidienne).

Théorème 1 : La suite $\{u^i\}$ définie par la relation (16) satisfait la relation :

$$\sup_i w_1(u^i) \geq \max_{u \in R^{2.n}} w_1(u) - \frac{1}{2} \cdot \bar{t} \cdot \limsup_{i \rightarrow \infty} \|v^k(u^i)\|^2.$$

Ce théorème montre bien l'importance du rôle du paramètre $\bar{t}$ dont nous parlerons à la fin du §2. Dans la pratique, le critère d'arrêt de l'algorithme est déterminé par le choix d'un entier naturel p défini comme suit : s'il existe un entier h tel que :

$$\max_{0 \leq i \leq p(h+1)-1} w_1(u^i) \leq \max_{0 \leq i \leq ph-1} w_1(u^i)$$

c'est à dire que si aucune augmentation de la fonction w_1 n'est enregistrée au bout de p itérations, alors on arrête la procédure. La meilleure solution à retenir sera le point u tel que

$$w_1(u) = \max_{0 \leq i \leq ph-1} w_1(u^i)$$

2.5 - Premier algorithme

Pas (0) - $i := 0$; définir u^0 par l'initialisation (1) ; calculer $v^k(u^0)$ et $w_1(u^0)$; poser $\ell := 0$, $u^* := u^0$ et $w_1^* := w_1(u^0)$.

Pas (1) - Faire $i := i+1$, $u^i := u^{i-1} + \bar{t} \cdot v^k(u^{i-1})$; calculer $v^k(u^i)$ et $w_1(u^i)$:
si $w_1(u^i) > w_1^*$ aller en (2) sinon aller en (3).

Pas (2) - Poser $u^* := u^i$; $w_1^* := w_1(u^i)$ et $\ell := 0$; aller en (1).

Pas (3) - Faire $\ell := \ell+1$; si $\ell \leq p$ alors aller en (1) sinon terminer : la solution à prendre est u^* .

2.6 - Description du deuxième algorithme

Il s'agit tout simplement d'une variante de la procédure précédente. Elle consiste à calculer une suite $\{u^i\}$ définie par la relation de récurrence suivante :

$$u^{i+1} = u^i + t_i \cdot v^k(u^i) \quad (17)$$

où $\{t_i\}$ est une suite de scalaires vérifiant :

- (i) $\lim_{i \rightarrow \infty} t_i = 0$
- (ii) $\sum_{i=0}^{\infty} t_i = +\infty$

Cette procédure est justifiée par le théorème de convergence suivant :

2.7 - Convergence de la deuxième procédure

Théorème 2 [7], [13] : Pour toute initialisation u^0 , la suite $\{u^i\}$ définie par la relation (17) admet une sous-suite $\{u^{k_i}\}$ qui converge vers une solution optimale u^* de la fonction concave w_1 .

En particulier, s'il existe un indice i tel que $u^{i+1} = u^i$, alors u^i est une solution optimale de la fonction w_1 .

Puisque la fonction w_1 est continue comme enveloppe convexe inférieure d'une famille de fonctions affines, la sous-suite des valeurs $\{w_1(u^{k_i})\}$ converge vers $w_1(u^*)$. Soit $\{w_1(u^{\ell_i})\}$ une sous-suite de $\{w_1(u^{k_i})\}$ qui vérifie la propriété suivante :

$$\forall i \in \mathbb{N}, \quad w_1(u^{\ell_i}) \leq w_1(u^{\ell_{i+1}}) \quad (18)$$

il est évident que la suite $\{w_1(u^{\ell_i})\}$ converge aussi vers $w_1(u^*)$.

Dans la pratique, on extrait une sous-suite $\{w_1(u^{\ell_i})\}$ de la suite initiale $\{w_1(u^i)\}$ qui vérifie la relation (18) et on considère qu'il y a convergence dès que :

$$|w_1(u^{\ell_i}) - w_1(u^{\ell_{i+1}})| \leq \varepsilon$$

où ε est un réel positif choisi d'avance.

2.8 - Deuxième algorithme

- Pas (0) - Poser $i := 0$; définir u^0 par initialisation (1) ; calculer $v^k(u^0)$ et $w_1(u^0)$: poser $u^* := u^0$, $w_1^* := w_1(u^0)$.
- Pas (1) - Faire $i := i+1$, $u^i := u^{i-1} + t_{i-1} \cdot v^k(u^{i-1})$; calculer $v^k(u^i)$ et $w_1(u^i)$; si $w_1(u^i) \geq w_1^*$ aller en (2), sinon aller en (1).
- Pas (2) - Poser $u^* := u^i$; si $|w_1(u^i) - w_1^*| \leq \epsilon$ terminer : la solution est u^* sinon poser $w_1^* := w_1(u^i)$ et aller en (1).

2.9 - Efficacité pratique

En pratique, les deux procédures de pénalité sont presque de même vitesse de convergence. Elles sont très sensibles aux choix des paramètres : $\bar{t}$ et p pour la première procédure et ϵ et la suite $\{t_i\}$ pour la seconde, choix qui dépendent de la nature du problème. Malheureusement, nous ne disposons pas de critères théoriques précis qui permettent de déterminer ces paramètres de manière optimale.

Le théorème 1 montre que l'efficacité de la première procédure de pénalité dépend beaucoup de la valeur de $\lim_{i \rightarrow \infty} \sup ||v^k(u^i)||^2$. Plus cette valeur est petite, plus la méthode est mieux adaptée au problème. Utilisée dans [6] pour résoudre le problème du voyageur de commerce, cette procédure a donné des résultats surprenants, compte tenu du fait que $||v^k(u^i)||^2$ converge souvent vers 0 quand i tend vers l'infini. Malheureusement, cette régularité ne se retrouve pas ici. En effet, d'après la proposition 6, on a :

$$v^k(u^i) = \theta^\sigma + (0, \dots, 0, -a, 0, \dots, 0, -a, 0, \dots, 0)$$

Or les composantes de θ^σ sont de l'ordre de $m+i$ avec $2 \leq i \leq n-1$ tandis que a est égal à $(2m+n)(n-1)/2$, ce qui implique que $||v^k(u^i)||^2$ demeure très grande en général. Cette "non régularité" sur les composantes de $v^k(u^i)$ fait qu'il est impossible de trouver un paramètre $\bar{t}$ qui convient à la procédure : si $\bar{t}$ convient aux petites composantes, il ne convient pas aux grandes et vice versa.

Bien qu'il ne soit commenté que pour la première procédure de pénalité l'inconvénient entraîné par la "non régularité" des composantes de $v^k(u^i)$ est le même que pour la deuxième procédure de pénalité. Il peut être aussi expliqué de la manière suivante :

En supprimant le domaine D , on introduit une contrainte redondante $A3.x = a$ qui, en termes de variables duales, devient en quelque sorte une pénalité quand $u \notin D$. En effet, si $u \notin D$, la quantité $\min_{i,j} \{c_{ij} - u_i - u_{n+j}\} \cdot (2m+n)(n-1)/2$ devient négative et diminue la valeur de $\langle u, \theta^0 \rangle$ pour donner la valeur de $w_1(u)$. Dans les deux procédures de pénalité, en suivant la direction du vecteur $v^{k(u^1)}$, le point u^{i+1} sort souvent du domaine D et on a du mal à y revenir.

Les expériences montrent que, en général, les procédures de pénalité ne sont pas efficaces, surtout pour les problèmes de grandes tailles.

§3 - METHODE DE SOUS-GRADIENT GENERALISEE

Dans le paragraphe précédent, nous avons vu que le fait d'ajouter la contrainte $A3.x = a$ n'a pas apporté de solution convenable au problème (P') et que, finalement, nous n'avons pas intérêt à sortir du domaine D à cause de la pénalité que nous allons subir.

Dans cette partie de l'exposé, nous nous proposons d'étudier une procédure très semblable à celle du paragraphe précédent, en restant toutefois dans le domaine D . Reprenons donc la fonction w définie dans le chapitre II par :

$$w : R^{2 \cdot n} \rightarrow \bar{R}, w(u) = \begin{cases} \min_{\sigma \in \mathcal{G}_{[2, n-1]}} \langle u, \theta^\sigma \rangle & \text{si } u \in D \\ -\infty & \text{sinon} \end{cases}$$

La procédure que nous allons proposer consiste à calculer une suite $\{u^i\}$ par la relation de récurrence suivante :

$$\begin{aligned} (i) \quad & u^0 \in D \\ (ii) \quad & u^{i+1} = u^i + t_i \cdot d^i \end{aligned} \quad (19)$$

où les suites $\{t_i\} \subset R$ et $\{d^i\} \subset R^{2 \cdot n}$ seront à déterminer.

3.1 - Description de la procédure théorique

Pour faire cette description, nous avons besoin de la définition suivante :

Définition 2 : Soit $v \in R^{2 \cdot n} - D$, nous appelons projection de v sur D et nous notons $P_D(v)$ le point $v' \in D$ vérifiant :

$$\forall v'' \in D, \quad ||v - v'|| \leq ||v - v''||.$$

Puisque D est convexe fermé, pour tout v , $P_D(v)$ existe et est unique [9].

Construction du vecteur d^i

Le vecteur d^i de la relation (19) dépend uniquement du point u^i . Pour simplifier les notations, nous supprimons l'indice i et construisons donc le vecteur d correspondant au point u .

Soit $u \in D$ et considérons le $2.n$ -vecteur θ^σ où σ est un élément quelconque de $\mathcal{G}(u)$. Désignons par $\gamma^1, \gamma^2, \dots, \gamma^p$ les éléments de $\Gamma(u)$ (voir chapitre II). Deux cas sont possibles :

1er cas : $\forall i \in \{1, 2, \dots, p\}, \langle \gamma^i, \theta^\sigma \rangle$ est non positif.

Dans ce cas, nous prenons $d = \theta^\sigma$.

2ème cas : $\exists i \in \{1, 2, \dots, p\}, \langle \gamma^i, \theta^\sigma \rangle$ est positif.

Si I est une partie de l'ensemble $\{1, 2, \dots, p\}$, nous désignons par D_I l'ensemble convexe $\{u / \forall i \in I, \langle u, \gamma^i \rangle \leq c_{kl} \text{ où } k \text{ et } n+l \text{ sont les rangs des composantes non nulles, égales à } 1, \text{ de } \gamma^i\}$. Puisque D est l'intersection finie de tous les demi-espaces fermés de forme $u_i + u_{n+j} \leq c_{ij}$ pour $i = 1, 2, \dots, n$ et $j = 1, 2, \dots, n$, il est facile de voir que :

$\exists I \subset \{1, 2, \dots, p\}, \exists \rho$ strictement positif assez petit tel que $P_D(u + \rho \cdot \theta^\sigma) = P_{D_I}(u + \rho \cdot \theta^\sigma)$. (Voir Figure 2).

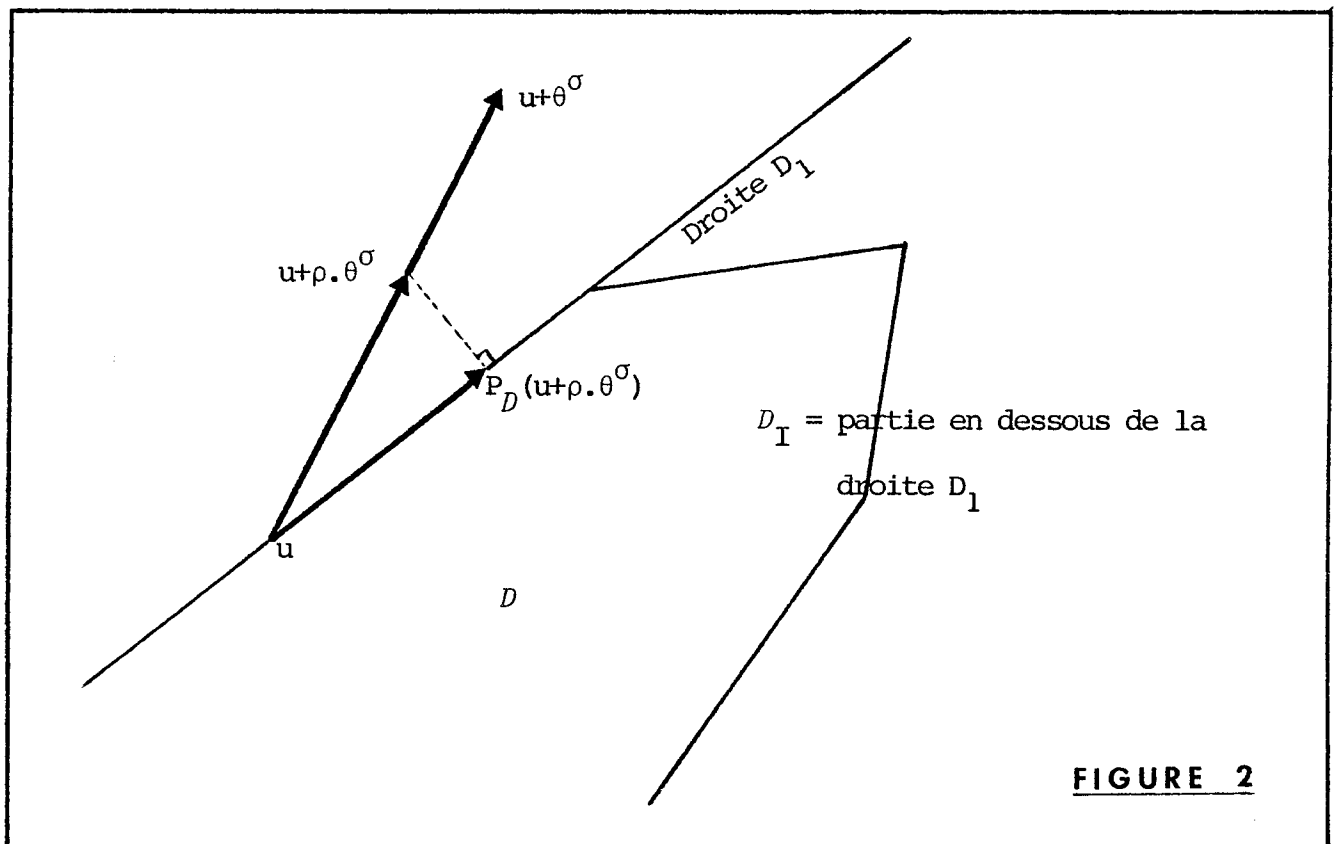


FIGURE 2

Dans ce cas, nous prenons $d = \frac{1}{\rho} (P_D(u + \rho \cdot \theta^\sigma) - u)$

Remarque 6 :

Il est clair que $u + \rho \cdot d \in D$. D'autre part, puisque $u = P_D(u)$, d'après un résultat dans [9], on a :

$$\begin{aligned} \|d\| &\leq \|u + \rho \cdot \theta^\sigma - u\| / \rho \\ &\leq \|\theta^\sigma\| \end{aligned}$$

Construction de la suite $\{t_i\}$

Soit $\bar{\epsilon}$ un réel positif fixé. D'après la construction de la direction d^i correspondant au point u^i , il existe $\rho > 0$ tel que $u^i + \rho \cdot d^i \in D$. Soit $\rho_i = \sup\{\rho / \rho > 0 \text{ et } u^i + \rho \cdot d^i \in D\}$, alors nous prenons $t_i = \min\{\bar{\epsilon}, \rho_i\}$. La construction de la suite $\{t_i\}$ étant faite, nous faisons l'hypothèse supplémentaire suivante :

(Hypo) : $\exists \epsilon > 0$ tel que $\forall i \quad t_i \geq \epsilon$.

Cette hypothèse signifie que, le point u et la direction d correspondante étant donnés, on peut toujours faire un pas de longueur supérieure ou égale à ϵ en restant dans D . Ceci est facilement vérifié dans la pratique.

3.2 - Convergence de la procédure théorique

Nous allons établir ici un résultat analogue au théorème 1 du paragraphe 2.

Lemme 1 : Soient u et $\bar{u}$ deux points de D tels que $w(\bar{u}) \geq w(u)$. Soit d le vecteur de direction calculé en 3.1 relatif au point u , alors

$$\langle \bar{u} - u, d \rangle \geq w(\bar{u}) - w(u) \geq 0.$$

Démonstration : Sans perte de généralité, on peut supposer que $u = 0$. Considérons le vecteur θ^σ à partir duquel est construit le vecteur d dans 3.1 d'après la proposition 7, on a :

$$\langle \bar{u}, \theta^\sigma \rangle \geq w(\bar{u}) - w(0) \geq 0 \quad (20)$$

D'après la construction du vecteur d , on a :

$$\rho \cdot d = P_D(\rho \cdot \theta^\sigma)$$

En appliquant un des résultats de [9], nous avons :

$$\langle \bar{u}, \rho \cdot d - \rho \cdot \theta^\sigma \rangle \geq 0$$

Ce qui implique que $\langle \bar{u}, d \rangle \geq \langle \bar{u}, \theta^\sigma \rangle$ et compte-tenu de (20), la démonstration s'achève. $\square$

Lemme 2 : Si $0 < t \leq \frac{2(w(\bar{u}) - w(u))}{||d||^2}$, alors $||\bar{u} - (u + t.d)|| \leq ||\bar{u} - u||$.

Démonstration : Même démonstration que le lemme 2 de [6]. $\square$

Notation 2 : Soit $\bar{w} \in \mathbb{R}$, nous désignons par $S_{\bar{w}}^-$ l'ensemble $\{u/w(u) \geq \bar{w}\}$.

Lemme 3 : Soit $\bar{w} < \max_{u \in D} w(u)$. Si la suite $\{u^i\}$ vérifie la relation de récurrence :

$$u^{i+1} = u^i + \lambda_i \frac{(\bar{w} - w(u^i))}{||d^i||^2} . d^i$$

alors, deux cas sont possibles :

1er cas : S'il existe ε_1 , $0 < \varepsilon_1 < \lambda_i \leq 2$ pour tout i , alors, soit il existe i tel que $u^i \in S_{\bar{w}}^-$, soit la suite $\{u^i\}$ converge vers un point appartenant à la frontière de $S_{\bar{w}}^-$.

2ème cas : Si $\lambda_i = 2$ pour tout i , alors il existe i tel que $u^i \in S_{\bar{w}}^-$.

Démonstration : Même démonstration que le lemme 3 de [6]. $\square$

Définition 3 : Soit $Q \subset \mathbb{R}^{2.n}$ et $\{\pi^i\}$ une suite à éléments dans $\mathbb{R}^{2.n}$. Nous disons que $\{\pi^i\}$ est Féjer-monotone par rapport à Q si pour tout $\pi \in Q$, la suite $||\pi^i - \pi||$ est monotone non croissante.

Un résultat connu dans [12] est le suivant :

Si $Q \neq \emptyset$ et si la suite $\{\pi^i\}$ est Féjer-monotone par rapport à Q , alors la suite $\{\pi^i\}$ est convergente.

Théorème 3 : La suite $\{u^i\}$ définie par la relation (19) satisfait à la propriété suivante :

$$\sup_i w(u^i) \geq \max_{u \in D} w(u) - \frac{1}{2} . \bar{t} . \limsup_{i \rightarrow \infty} ||d^i||^2$$

Démonstration : Procédons par l'absurde.

Soit $L = \limsup_{i \rightarrow \infty} ||d^i||^2$ et supposons que, pour tout i , on a :

$$w(u^i) < \hat{w} < \max_{u \in D} w(u) - \frac{1}{2} . \bar{t} . L \quad (21)$$

Soit $\bar{w} = \hat{w} + \frac{1}{2} \cdot \bar{t} \cdot L$ et considérons le lemme 3, la condition $\lambda_i \leq 2$ devient :

$$\frac{t_i \cdot ||d^i||^2}{\bar{w} - w(u^i)} \leq 2$$

ou encore

$$t_i \cdot ||d^i||^2 \leq 2 \cdot (\hat{w} - w(u^i)) + \bar{t} \cdot L$$

D'après (21) et par définition de L , cette inégalité est vérifiée à partir d'un certain rang i .

Pour appliquer le lemme 3, il suffit alors de vérifier que, pour tout i ,

$$\lambda_i = \frac{t_i \cdot ||d^i||^2}{\bar{w} - w(u^i)}$$

est supérieur à ε_0 où ε_0 est un réel positif fixé d'avance. Par hypothèse (Hypo), il existe $\varepsilon > 0$ tel que $\forall i, t_i \geq \varepsilon$. Il suffit alors de vérifier que :

(i) $\exists \varepsilon_1 > 0$ tel que $\forall i, ||d^i|| \geq \varepsilon_1$.

(ii) La suite $\{w(u^i)\}$ est bornée.

Montrons (ii). Puisque $\forall i, t_i \geq \varepsilon > 0$ et que $\lambda_i \leq 2$ à partir d'un certain rang i , on peut appliquer le lemme 2 et la suite $\{u^i\}$ est Féjer-monotone par rapport à S_w^- . Ce qui implique $\{u^i\}$ est bornée et il en est de même que $w(u^i)$. Soit $t > 0$ tel que $\forall i, ||u^i|| \leq t$. Montrons (i). Supposons que $\forall \varepsilon_1 > 0, \exists i$ tel que $||d^i|| < \varepsilon_1$. Soit u^* la solution optimale de la fonction $w : w(u^*) = \max_{u \in D} w(u)$. Prenons

$\varepsilon_1 = \frac{w(u^*) - \hat{w}}{t \cdot ||u^*||}$, $\exists i$ tel que $||d^i|| < \varepsilon_1$. D'après le lemme 1, on a :

$$\begin{aligned} w(u^*) - w(u^i) &\leq \langle u^* - u^i, d^i \rangle \\ &\leq ||u^* - u^i|| \cdot ||d^i|| \\ &< w(u^*) - \hat{w} \end{aligned}$$

D'où $w(u^i) > \hat{w}$, contradiction.

Donc, les hypothèses du lemme 3 sont satisfaites, alors que sa conclusion reste non vérifiée compte-tenu de la relation (21). $\square$

3.3 - Description de la procédure pratique

En pratique, la construction de la suite $\{t_i\}$ reste la même, seule celle de la suite $\{d^i\}$ subit une légère modification. En effet, pour $\forall i \in \mathbb{R}^{2 \cdot n - D}$, il est

difficile de calculer le point $P_D(v)$ dans $R^{2.n}$ rapporté à sa base canonique. Donc, étant donné un point $u \in D$, pour calculer le vecteur de direction d relatif au point u , nous nous proposons de travailler dans $R^{2.n}$ rapporté, non pas à sa base canonique mais à une base que l'on précisera.

Construction pratique de la suite $\{d^i\}$

Pour simplifier les notations, nous supprimons l'indice i . Soit $u \in D$ et construisons le vecteur de direction d relatif au point u . Soient $\gamma^1, \gamma^2, \dots, \gamma^{n.n}$ les vecteurs définissant le domaine D , c'est à dire que $D = \{u / \langle u, \gamma^i \rangle \leq c_{kl} ; i = 1, 2, \dots, n.n ; k \text{ et } n+l \text{ désignent les rangs des 2 composantes non nulles de } \gamma^i\}$. Supposons de plus $\Gamma(u) = \{\gamma^1, \gamma^2, \dots, \gamma^p\}$ avec $p \leq n.n$ et considérons le vecteur θ^σ où σ est un élément quelconque de $\mathcal{G}(u)$.

D'après le théorème des alternatives, soit le système :

$$(S) \begin{cases} \alpha \geq 0 \\ \Gamma(u) \cdot \alpha = \theta^\sigma \end{cases}$$

admet une solution auquel cas u est une solution optimale du problème (P'), soit il existe d vérifiant $d \cdot \Gamma(u) \leq 0$ et $\langle d, \theta^\sigma \rangle > 0$.

Le système (S) est très facile à résoudre compte-tenu de la forme particulière de la matrice $\Gamma(u)$.

Proposition 9 : La matrice $\Gamma(u)$ est totalement unimodulaire.

Démonstration : On peut la considérer comme une matrice d'incidence aux arêtes d'un graphe [15]. $\square$

Soit Γ une matrice d'ordre $2n \times 2.n$ telle que :

$$\Gamma \cdot \Gamma(u) = \bar{\Gamma}(u) \quad (22)$$

où $\bar{\Gamma}(u)$ est une matrice d'ordre $2.n \times p$ dont les vecteurs colonnes ne possèdent qu'une seule composante non nulle, égale à 1. On peut supposer que :

$$\bar{\Gamma}(u) = \begin{pmatrix} 1 & & & & & \\ & 1 & & & & \\ & & \ddots & & & \\ & & & 0 & & \\ & 0 & & & \ddots & \\ & & & & & \ddots \\ \hline & & & & & 1 \\ & & 0 & & & \end{pmatrix}$$

$\updownarrow$
 p_i

$\updownarrow$
 $2 \cdot n$

sinon, on fait une permutation de lignes et de colonnes pour la ramener à cette forme.

C'est la matrice Γ qui joue le rôle de matrice de changement de bases de $R^{2 \cdot n}$ pour les vecteurs γ^i , $i = 1, 2, \dots, n \cdot n$ d'une part et θ^σ d'autre part. On a alors les formules de changement de bases suivantes :

$$\hat{\gamma}^i = \Gamma \cdot \gamma^i \iff \gamma^i = \Gamma^{-1} \cdot \hat{\gamma}^i \quad \text{pour } i = 1, 2, \dots, n \cdot n \quad (23)$$

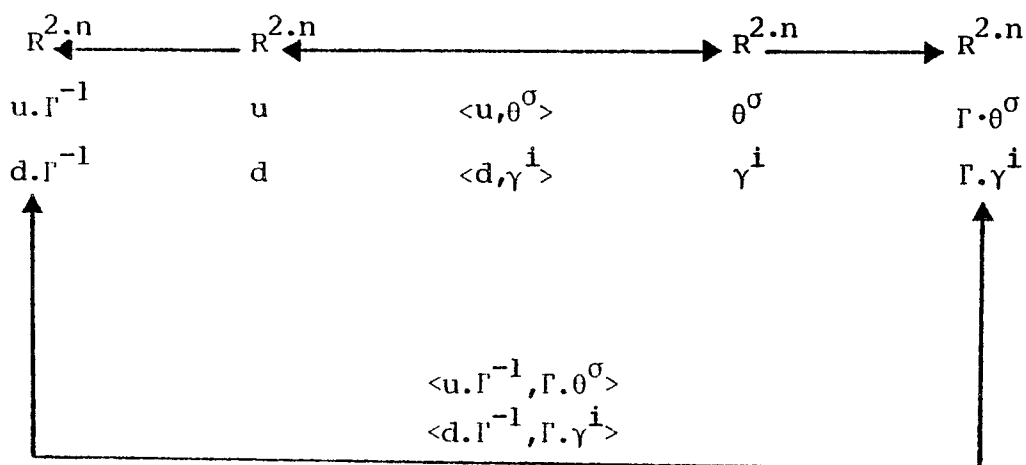
$$\hat{\theta}^\sigma = \Gamma \cdot \theta^\sigma \iff \theta^\sigma = \Gamma^{-1} \cdot \hat{\theta}^\sigma \quad (24)$$

De même, dans l'espace $R^{2 \cdot n}$ dual, les formules de changement de bases pour u et d sont :

$$\hat{u} = u \cdot \Gamma^{-1} \iff u = \hat{u} \cdot \Gamma \quad (25)$$

$$\hat{d} = d \cdot \Gamma^{-1} \iff d = \hat{d} \cdot \Gamma \quad (26)$$

Cette transformation est illustrée par le diagramme suivant :



Cette transformation étant faite, nous pouvons calculer la direction d relative au point u en faisant une projection dans les espaces transformés.

En effet, le point u étant donné, considérons le vecteur θ^σ où σ est un élément quelconque de $\mathcal{G}(u)$. D'après (24), $\hat{\theta} = \Gamma \cdot \theta^\sigma$. Considérons le vecteur d défini à partir de $\hat{\theta}$ de la façon suivante :

$$\hat{d}_i = \begin{cases} \hat{\theta}_i & \text{si } \hat{\theta}_i < 0 \text{ et } 1 \leq i \leq p \\ 0 & \text{si } \hat{\theta}_i \geq 0 \text{ et } 1 \leq i \leq p \\ \hat{\theta}_i & \text{si } p+1 \leq i \leq 2.n \end{cases} \quad (27)$$

Proposition 10 : Soit $\hat{D} = \{\hat{v}/\hat{v} = u \cdot \Gamma^{-1}, u \in D\}$ le transformé de D , alors le vecteur $\hat{d}$ n'est autre que la projection du vecteur $\hat{\theta}$ sur $\hat{D}$ et il existe $\rho > 0$ tel que $\hat{u} + \rho \cdot \hat{d} \in \hat{D}$.

Démonstration : Evidente d'après la construction de $\hat{d}$. $\square$

Une telle direction $\hat{d}$ relative au point $\hat{u}$ étant déterminée, pour avoir la direction d relative au point u , il suffit d'appliquer la relation (26) et on a :

$$d = \hat{d} \cdot \Gamma.$$

En résumé, un point $u \in D$ étant donné, pour calculer d , on calcule d'abord Γ telle que $\Gamma \cdot \Gamma(u) = \bar{\Gamma}(u)$, puis $\hat{\theta} = \Gamma \cdot \theta^\sigma$. De $\hat{\theta}$, on déduit $\hat{d}$ en projetant $\hat{\theta}$ sur $\hat{D}$ par la relation (27), puis on calcule $d = \hat{d} \cdot \Gamma$. Remarquons enfin que la projection faite ici est la projection dans $R^{2.n}$ rapporté, non pas à sa base canonique, mais à ses bases définies par les matrices Γ et Γ^{-1} .

Remarque 7 : Bien que la projection du vecteur θ^σ ne soit pas faite dans $R^{2.n}$ rapporté à sa base canonique, nous sommes convaincus que le résultat du théorème 3 reste vrai pour la suite $\{u^i\}$ ainsi calculée. Puisque ceci ne se situe pas dans l'orientation de notre travail, nous le laissons comme problème ouvert.

Critère d'arrêt : C'est le même critère que la première procédure de la méthode de pénalité. Il est défini par un entier p de la façon suivante : s'il existe un entier h tel que :

$$\max_{0 \leq i \leq p(h+1)-1} w(u^i) \leq \max_{0 \leq i \leq ph-1} w(u^i)$$

c'est à dire que si aucune augmentation de la fonction w n'est enregistrée au bout de p itérations, alors on arrête l'opération et la solution à retenir sera le point u^* tel que

$$w(u^*) = \max_{0 \leq i \leq p-1} w(u^i)$$

3.4 - Algorithme

A chaque étape i de l'algorithme, la variable courante u sera notée par u^i , Γ par Γ^i , $\hat{\theta}$ par $\hat{\theta}^i$ et $\hat{d}$ par $\hat{d}^i$.

Pas (0) - $i := 0$; définir u^0 par l'initialisation (1) ; calculer $\Gamma(u^0)$, Γ^0 , $\hat{\theta}^0$, $\hat{d}^0$, d^0 , t_0 et $w(u^0)$; poser $\ell := 0$, $u^* := u^0$ et $w^* := w(u^0)$.

Pas (1) - Faire $i := i+1$, $u^i := u^{i-1} + t_{i-1} \cdot d^{i-1}$; calculer $\Gamma(u^i)$, Γ^i , $\hat{\theta}^i$, $\hat{d}^i$, d^i , t_i et $w(u^i)$; si $w(u^i) > w^*$ aller en (2) sinon aller en (3).

Pas (2) : Poser $u^* := u^i$, $w^* := w(u^i)$ et $\ell := 0$; aller en (1).

Pas (3) - Faire $\ell := \ell+1$; si $\ell \leq p$ alors en aller en (1) sinon terminer : la solution à prendre est u^* .

3.5 - Efficacité pratique

La réussite pratique de cette procédure provient :

(i) De la forme particulière de la matrice $\Gamma(u)$ pour un point $u \in D$ donné. L'unimodularité de la matrice $\Gamma(u)$ implique en outre que le vecteur direction d relatif au point u est entier, ce qui permet d'éviter les erreurs de précision rencontrées au §1. En plus de l'unimodularité, il est très facile "d'inverser" la matrice $\Gamma(u)$ dont les colonnes ne possèdent que deux composantes non nulles, égales à 1. Nous sommes convaincus qu'une sérieuse optimisation en programmation permet d'améliorer davantage la performance de la procédure.

(ii) Du fait qu'à chaque itération, la variable courante u reste toujours dans le domaine D et donc que la valeur de la fonction w augmente pratiquement à chaque itération. Ce phénomène est d'ailleurs confirmé par la pratique.

Les expériences pratiques montrent aussi qu'il n'est pas moins intéressant de considérer les deux variantes suivantes de la méthode :

1ère variante : Garder la matrice inverse de base Γ à chaque itération. En effet, souvent dans la pratique, on remarque que d'une itération où la variable courante est u^i à la suivante où la variable courante est u^{i+1} , $\Gamma(u^i)$ et $\Gamma(u^{i+1})$ possèdent à peu près les mêmes vecteurs colonnes.

2ème variante : Garder la direction d^i quand la fonction w subit un accroissement assez important. Plus précisément, soit d^i la direction relative au point u^i et t un réel positif fixé qui dépend du problème. Si $w(u^i + t \cdot d^i) \geq w(u^i) + t$, alors on considère que d^i est la direction relative au point u^{i+1} .

Il convient de remarquer qu'on peut très bien appliquer les deux variantes en même temps.

§4 - TABLEAUX DE COMPARAISON

Une version de chacune des méthodes est programmée en Algol W et essayée sur l'IRIS 80, les coûts sont soit les nombres gérés aléatoirement entre 0 et 100, soit les distances entre les points gérés aléatoirement dans un carré de 50 unités de côtés. Le temps de calcul est exprimé en 60^e de seconde. Pour simplifier, la méthode ascendante, la première procédure de pénalité, la seconde procédure de pénalité et la méthode de sous-gradient généralisée seront appelées respectivement méthode 1, 2, 3 et 4. Pour la méthode 3, la suite $\{t_i\}$ vérifie la relation $t_i = 1/(t_1 + i)$ et ne dépend donc que de t_1 . Pour ces quatre méthodes, on n'enregistre la valeur de w que lorsqu'il y a accroissement. Ces quatre méthodes sont en outre comparées avec un simplexe standard appliqué sur le programme (PC).

Premier essai

$n = 7$; $m = 6$; la valeur de la solution optimale est $w^* = 140$; le simplexe prend 288 unités de temps ; pour la méthode 2, $p = 50$, $\bar{t} = 2/3(2m+n)(n-1)$; pour la méthode 3, $\epsilon = 10^{-5}$, $t_1 = 3(2m+n)(n-1)$; pour la méthode 4, $p = 50$, $\bar{t} = 2/(2m+n)$. Le tableau 1 donne les valeurs de la fonction objective en fonction du temps.

Temps	Méthode 1 : Fonction objective	Méthode 2 : Fonction objective	Méthode 3 : Fonction objective	Méthode 4 : Fonction objective
4	107	107	107	107
5				
10				
12		114		
13		117	117	
16	118	125		133
19				
21			123	
31		129		
38			129	
42	136	132		
77			131	
85				
88				
103			132	
213	140	134		

TABLEAU 1

Deuxième essai

$n = 9$; $m = 6$; la valeur de la solution optimale est $w^* = 680$; le simplexe prend 1368 unités de temps ; pour la méthode 2, $p = 50$, $\bar{t} = 1/(2m+n)(n-1)$; pour la méthode 3, $\varepsilon = 10^{-5}$, $t_1 = (2m+n)(n-1)$; pour la méthode 4, $p = 50$, $\bar{t} = 1/(2m+n)$. Le résultat est résumé par le tableau 2.

Temps	Méthode 1 : Fonction objective	Méthode 2 : Fonction objective	Méthode 3 : Fonction objective	Méthode 4 : Fonction objective
6		488	488	
15				488
18			504	533
23		502	511	
28	488		528	549
34	493	512		
40			543	590
44			562	
47	525	525		633
58	617	537		
62			588	
67		551	599	657
82			614	
89			620	
98		581		
108	660	593	644	662
122	678			666
144	680	626	655	
165				680
177		644	660	
182		654		
205			664	
207		659		

TABLEAU 2

Troisième essai

$n = 15$; $m = 2$; le simplexe a pris plus de 5 minutes sans arriver à l'optimum ; la méthode 1 n'est pas applicable à cause des erreurs de précision ; pour la méthode 2, $p = 50$, $\bar{t} = 2/3(2m+n)(n-1)$; pour la méthode 3, $\epsilon = 10^{-5}$, $t_1 = 3(2m+n)(n-1)/2$; pour la méthode 4, $p = 50$, $\bar{t} = 1/(2m+n)$. Le résultat est résumé par le tableau 3.

Quatrième essai

$n = 15$; $m = 10$; le simplexe a pris plus de 5 minutes sans arriver à l'optimum ; la méthode 1 n'est pas applicable à cause des erreurs de précision ; pour la méthode 2, $p = 50$, $\bar{t} = 2/3(2m+n)(n-1)$; pour la méthode 3, $\epsilon = 10^{-5}$, $t_1 = 3(2m+n)(n-1)/2$; pour la méthode 4, $p = 50$, $\bar{t} = 1/(2m+n)$. Le résultat est résumé par le tableau 4.

Temps	Méthode 2 : Fonction objective	Méthode 3 : Fonction objective	Méthode 4 : Fonction objective
12	697	697	
34			697
37			756
61			796
70			837
78		714	
82	705		
100		725	839
105	716		
113		728	
123	718		856
140		749	
145	726		
152			863
172	747	756	
198		761	
200	750	766	875
224		776	
232	759		
258	760	780	
272	769		
279		786	
295	771	790	885
355	791	802	
381		807	
392	801		

TABLEAU 3

Temps	Méthode 2 : Fonction objective	Méthode 3 : Fonction objective	Méthode 4 : Fonction objective
11	2496	2496	
32			2496
35			2635
63	2508	2508	2643
89	2520	2521	
96	2535		
127	2548	2562	
145	2775		
159		2592	2649
176	2584		
196			2673
223	2608		
218		2611	
239	2625		
277	2654	2625	
291		2634	
310		2653	
352		2658	
		2658	

TABLEAU 4

CHAPITRE IV

METHODES DE SEPARATION
ET
EVALUATION PROGRESSIVE

§1 - INTRODUCTION

Comme nous l'avons indiqué, la formulation du problème de ramassage scolaire (P) en programme linéaire continu (PC) ne permet pas de résoudre complètement le problème (P). Ce programme continu fournit seulement une évaluation par défaut du coût d'une tournée optimale. Plus l'évaluation est bonne, c'est à dire que plus la borne fournie est proche du coût d'une tournée optimale, plus la formulation est dite efficace. Il est très difficile de définir des critères théoriques pour mesurer l'efficacité de telle ou telle formulation. Une manière d'apprécier cette efficacité est de développer, en parallèle avec la formulation, une méthode de Séparation et Evaluation Progressive (notée dans la suite par SEP) [5], [16] qui permet d'achever la résolution complète du problème de départ. La qualité de la formulation est alors jugée par le nombre de solutions explorées par la procédure de SEP ; plus ce nombre est petit, meilleure est la formulation. Il convient enfin de remarquer que ce nombre, appelé dans la suite le nombre de sommets de l'arborescence explorés, dépend lui aussi de la méthode SEP choisie.

Dans ce chapitre, nous proposons deux procédures de séparation de l'ensemble des solutions réalisables. La première, moins efficace, est pourtant la plus naturelle.

§2 - PREMIER ALGORITHME DE SEP

2.1 - Description générale

Une procédure de SEP est caractérisée par une arborescence d'exploration. Chaque sommet i de l'arborescence correspond à un sous-ensemble R_i de l'ensemble des solutions réalisables -ici les tournées- muni d'une borne inférieure du coût d'une tournée optimale dans R_i . A la racine de l'arborescence correspond l'ensemble R_0 de toutes les tournées, muni de la borne inférieure fournie par le programme linéaire continu (PC). Si un sommet pendant i de l'arborescence dont le sous-ensemble de tournées est R_i , muni de la borne b_i , est choisi comme "noeud de séparation", R_i est alors partitionné en $R_{i_1}, R_{i_2}, \dots, R_{i_k}$ munis de bornes respectives $b_{i_1}, b_{i_2}, \dots, b_{i_k}$. Plus précisément, on a :

$$R_i = R_{i_1} \cup R_{i_2} \cup \dots \cup R_{i_k}$$

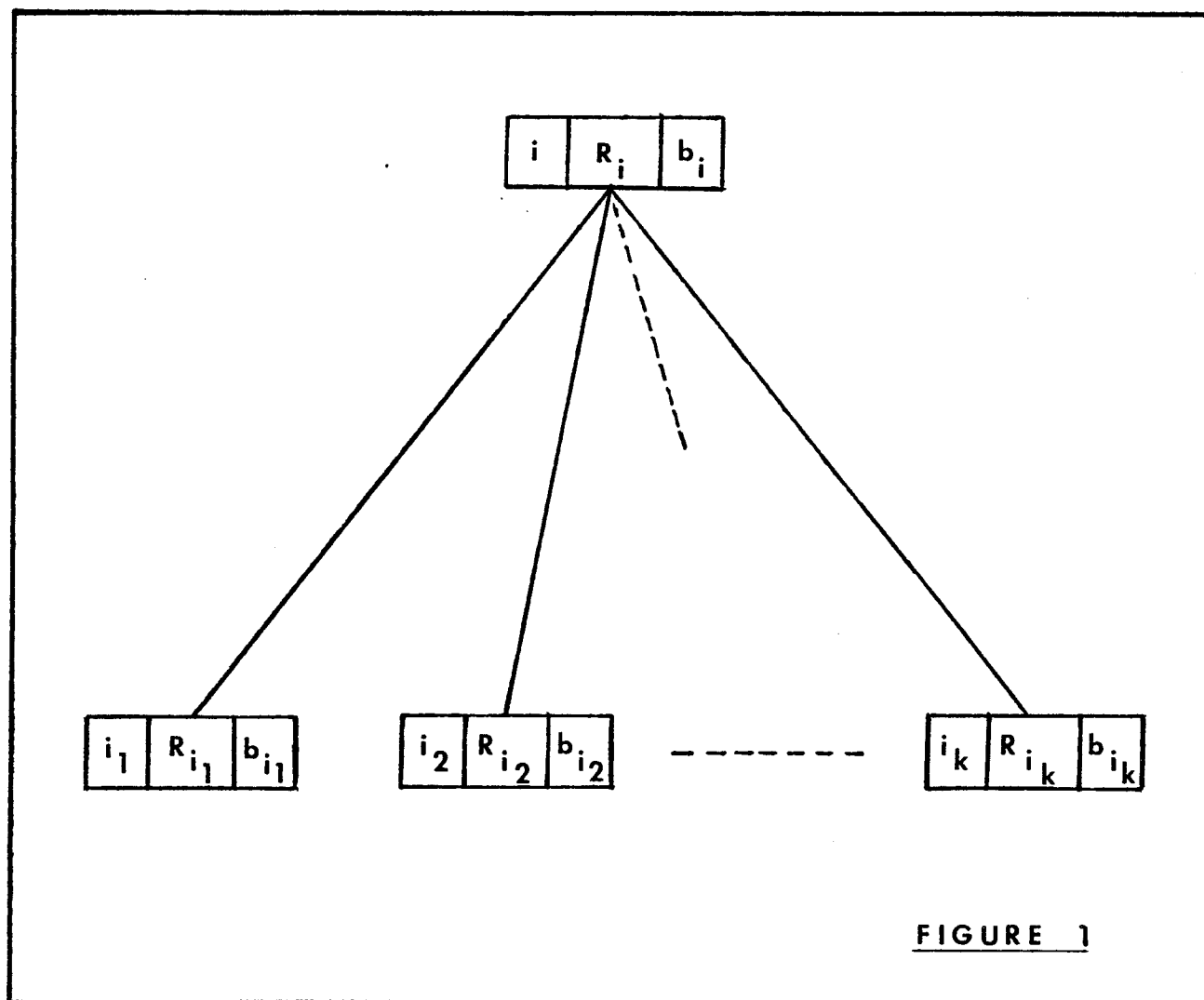
$$R_{i_p} \cap R_{i_q} = \emptyset \text{ pour } 1 \leq p < q \leq k$$

et
$$b_{i_p} \leq \min\{c(T)/T \in R_{i_p}\}$$

Le sommet i devient alors non pendent et donne naissance à k sommets pendants. En schématisant chaque sommet i par le triplet (i, R_i, b_i) , cette séparation est illustrée par la figure 1.

Définition 1 : On appelle sommet terminal tout sommet pendent i tel que $b_i = +\infty$ ou tel qu'il existe $T \in R_i$ et $b_i = c(T)$.

On verra plus loin que, selon un certain critère de choix que nous adopterons, un sommet terminal ne sera jamais séparé et restera toujours sommet pendent jusqu'à la fin de la procédure.



Les deux procédures de SEP que nous proposons ici comprennent chacune cinq étapes principales suivantes :

Etape 1 - Choix du noeud de séparation.

Etape 2 - Partitionnement du sous-ensemble des tournées.

Un sommet pendant (i, R_i, b_i) étant choisi comme noeud de séparation, il faut définir une stratégie qui permet de partitionner le sous-ensemble des tournées R_i en $R_{i_1}, R_{i_2}, \dots, R_{i_k}$. C'est justement à cette étape que les deux procédures de SEP diffèrent l'une de l'autre. Les quatre autres étapes restent communes aux deux procédures.

Etape 3 - Reconstruction des données.

Après partitionnement, le sommet pendant i donne naissance à k sommets pendants $i_1, i_2, \dots, i_k$ avec les sous-ensembles de tournées $R_{i_1}, R_{i_2}, \dots, R_{i_k}$ respectivement. Pour calculer les bornes inférieures $b_{i_1}, b_{i_2}, \dots, b_{i_k}$ associées, nous construisons, pour chaque i_p et R_{i_p} , un nouvel entier naturel n_{i_p} (le nombre de sommets du "graphe réduit"), un nouvel entier naturel m_{i_p} ($m_{i_p} + 1$ représente le nombre de personnes au départ dans le graphe réduit) et un nouveau système de coûts c_{i_p} .

Etape 4 - Calcul d'une borne.

Cette étape consiste à calculer une borne b_i par le programme linéaire (PC) construit à partir du réseau réduit donné par l'étape 3. Il est clair que, si $n_{i_p} = 3$, alors b_{i_p} est donnée, non par le programme linéaire (PC), mais par la valeur du coût de la tournée triviale sur un graphe de trois sommets. Dans ce cas, on verra plus loin comment le sommet i_p devient un sommet terminal.

Etape 5 - Détermination d'une tournée optimale.

Cette étape n'intervient qu'à la fin de la procédure de SEP. Elle consiste à déterminer une tournée optimale dans le réseau de départ à partir d'une tournée optimale dans le réseau réduit.

2.2 - Description détaillée de l'Etape 1

Nous adoptons ici le choix classique qui consiste à prendre comme noeud de séparation un sommet pendant de l'arborescence dont la borne associée est la plus petite parmi les sommets pendants. Un résultat bien connu relatif à ce choix est le suivant :

Proposition 1 [5] : Avec cette stratégie de choix, si le sommet (i, R_i, b_i) de l'arborescence pris comme noeud de séparation est terminal, alors la tournée T^* telle que $c(T^*) = b_i$ est optimale.

Cette proposition explique aussi pourquoi dans la résolution d'un problème de minimisation par la méthode SEP, on calcule les bornes inférieures associées aux sommets de l'arborescence explorés.

2.3 - Description détaillée de l'Etape 2

Nous faisons d'abord la description sur la racine (o, R_o, b_o) de l'arborescence. Pour tout autre sommet pendant quelconque (i, R_i, b_i) , le partitionnement se fait d'une manière presque identique.

Etape 2 pour la première méthode de SEP - Comme nous l'avons indiqué (Remarque 1, Chapitre I), l'ensemble des tournées est en bijection avec l'ensemble des permutations $\mathcal{G}_{[2, n-1]}$. A chaque permutation $\sigma = (\sigma_2, \sigma_3, \dots, \sigma_{n-1})$ de $\mathcal{G}_{[2, n-1]}$ correspond une tournée $T = (1, \sigma_2, \sigma_3, \dots, \sigma_{n-1}, n)$ et réciproquement. Rappelons nous que, pour calculer b_o , nous avons à résoudre le problème :

$$(P') : \max_{u \in D} w(u) \quad \text{où} \quad w(u) = \min_{\sigma \in \mathcal{G}_{[2, n-1]}} \langle u, \theta^\sigma \rangle$$

Les résultats pratiques sur les petits exemples ont montré que, si $u^* = (u_1^*, u_2^*, \dots, u_{2n}^*)$ désigne une solution du problème (P') calculée par l'une des méthodes proposées dans le chapitre III, alors la permutation $\sigma^* \in \mathcal{G}(u^*)$ correspond souvent à une tournée $T = (1, \sigma_2^*, \sigma_3^*, \dots, \sigma_{n-1}^*, n)$ qui est "non loin" de la tournée optimale. Presque toujours, dans la tournée optimale, on trouve le sommet σ_2^* en deuxième position. Ces remarques ont conduit à la séparation la plus naturelle de l'ensemble R_o en deux sous-ensembles R_{o1} et R_{o2} : R_{o1} est l'ensemble des tournées qui n'admettent pas le sommet σ_2^* en deuxième position et R_{o2} est l'ensemble des tournées qui admettent obligatoirement le sommet σ_2^* en deuxième position.

Etape 2 pour la deuxième méthode de SEP - Il s'agit ici d'un partitionnement dont la forme n'est pas loin du premier. Soit u^* une solution calculée par l'une des méthodes proposée dans le chapitre III pour résoudre le problème (P'). Soit q un entier vérifiant :

$$2 \leq q \leq n-1 \text{ et } c_{1q} - u_1^* - u_{n+q}^* = \min_{2 \leq i \leq n-1} \{c_{1i} - u_1^* - u_{n+i}^*\} \quad (1)$$

Le partitionnement consiste alors à subdiviser R_0 en deux sous-ensembles R_{01} et R_{02} : R_{01} est l'ensemble des tournées qui n'admettent pas le sommet q en deuxième position et R_{02} est l'ensemble des tournées qui admettent obligatoirement le sommet q en deuxième position.

Remarque 1 : On peut supposer que l'entier q défini par (1) vérifie la relation $c_{1q} - u_1^* - u_{n+q}^* = 0$ sinon on pose :

$$\bar{u} = (c_{1q} - u_{n+q}^*, u_2^*, \dots, u_{2n}^*)$$

et il est clair que $\bar{u} \in D$ et $w(\bar{u}) > w(u^*)$.

Remarque 2 : Les principes des deux partitionnements sont très différents. Le premier cherche à "découvrir" le plus vite possible une tournée optimale. En effet, si σ_2^* est bien en deuxième position dans la tournée optimale, la borne b_{02} associée à R_{02} restera inférieure au coût d'une tournée optimale. D'autre part, nous espérons que le fait d'exclure σ_2^* de la deuxième position augmente beaucoup la valeur de la borne b_{01} associée à R_{01} . S'il en est ainsi, le critère de choix à l'Etape 1 conduirait alors à explorer toujours dans la bonne branche de l'arborescence où se trouverait une tournée optimale. Malheureusement, les résultats pratiques ont confirmé le contraire. Ils ont montré que le fait d'exclure σ_2^* de la deuxième position n'augmente pratiquement pas la valeur de b_{01} et que la procédure a tendance à explorer dans cette mauvaise branche de l'arborescence. Nous parlerons du deuxième partitionnement dans l'Etape 4 où nous aurons plus d'éléments pour le commenter.

Remarque 3 : Nous pouvons aussi envisager les partitionnements classiques qui consistent à mettre certains arcs du réseau dans les tournées et à en exclure certains autres, mais nous ne le faisons pas pour les deux raisons suivantes :

(i) Cette façon de faire ne permet pas de fixer la valeur du trafic sur les arcs à prendre comme dans le cas où les variables de trafics x_{ij} sont bi-valentes, alors que pour notre partitionnement, le fait de mettre le sommet q en deuxième position fixe le trafic sur l'arc $(1,q)$ égal à $m+1$.

(ii) Le fait de mettre un sommet q en deuxième position permet de grouper le sommet 1 et q pour en faire un seul sommet dans le réseau réduit et de réduire ainsi la taille du problème.

2.4 - Description détaillée de l'Etape 3

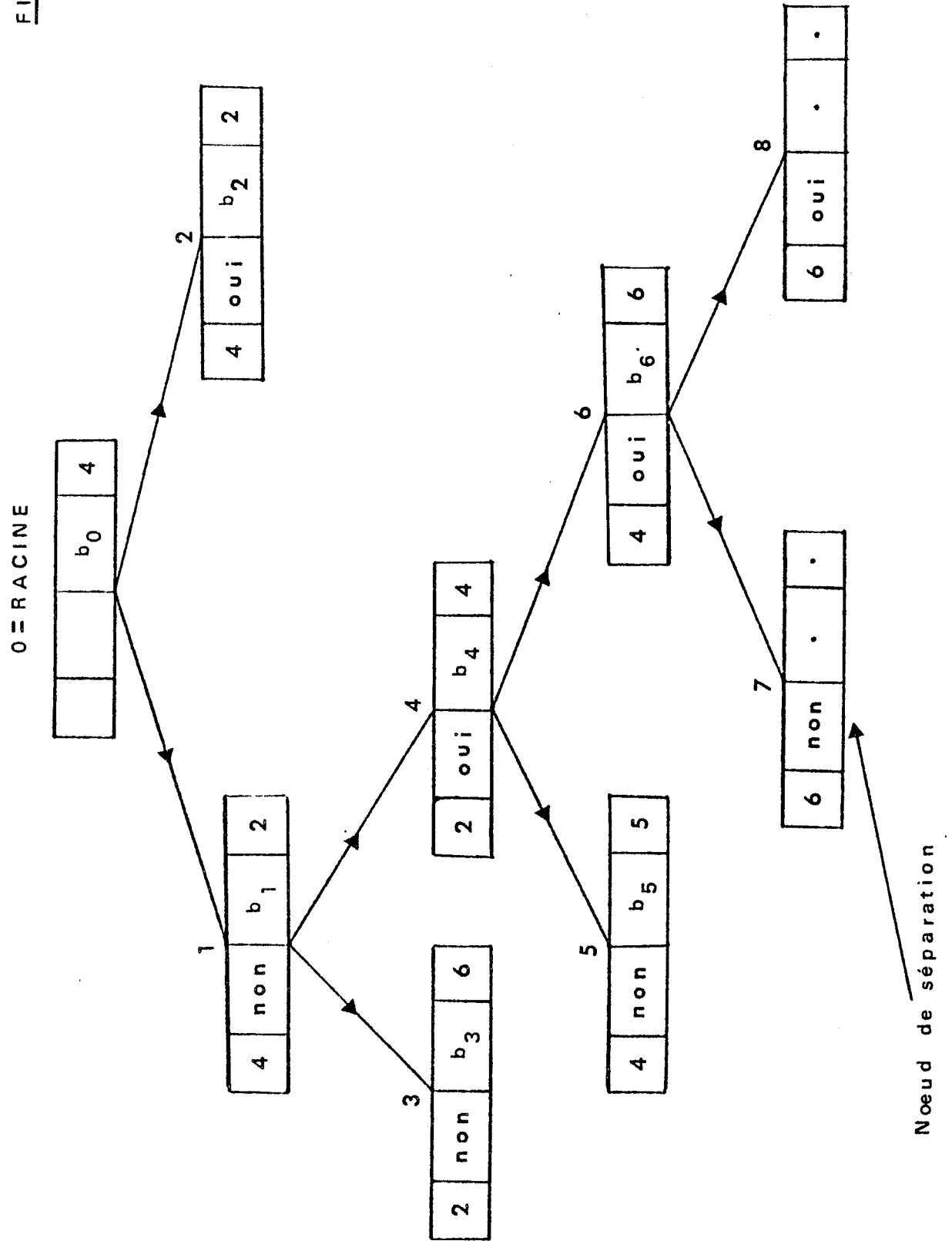
Définition 2 : Si, après partitionnement de l'ensemble des tournées de R_1 , un sommet i de l'arborescence donne naissance à k nouveaux sommets $i_1, i_2, \dots, i_k$, on dit que le sommet i est le père des sommets $i_1, i_2, \dots, i_k$ et que les sommets $i_1, i_2, \dots, i_k$ sont les fils du sommet i .

La façon de partitionner décrite à l'Etape 2 nous fait remarquer que :

- (i) Tout sommet de l'arborescence possède exactement deux fils.
- (ii) Tout sommet i de l'arborescence peut être caractérisé par un quadruple (k, ℓ, b, p) où
 - k est un entier donné par le calcul de la borne du père du sommet i décrit à l'Etape 2. Plus précisément, dans le partitionnement de l'ensemble des tournées associé au père du sommet i , l'entier k joue le rôle de σ_2^* où q décrit à l'Etape 2.
 - ℓ est une variable logique qui peut prendre deux valeurs non ou oui, $\ell = \text{non}$ signifie que les tournées associées au sommet i de l'arborescence n'admettent pas k en deuxième position, $\ell = \text{oui}$ signifie que les tournées associées à i admettent k en deuxième position.
 - b est un réel égal à la borne associée au sommet i de l'arborescence. Son calcul sera décrit à l'Etape 4.
 - p est un entier jouant le même rôle par rapport au sommet i de l'arborescence que k par rapport au père de i .

A la racine de l'arborescence, l'entier k et la variable logique ℓ peuvent prendre des valeurs arbitraires. Cette description est illustrée par la figure 2 où $n = 7$ par exemple et où le numéro des sommets de l'arborescence est écrit à côté, à l'extérieur des quatre cases caractérisant les sommets.

FIGURE 2



Notation 1 : Dans la suite, un sommet i de l'arborescence sera noté par (k_i, ℓ_i, b_i, p_i) . Avant le calcul de b_i , donc aussi de p_i , il sera noté par $(k_i, \ell_i, \dots)$.

Etant donné un sommet $i = (k_i, \ell_i, \dots)$ de l'arborescence, pour pouvoir calculer b_i et p_i , il faut connaître :

- (i) n_i , le nombre de sommets du graphe réduit constituant le réseau réduit.
- (ii) m_i tel que m_i+1 représente le nombre de personnes au départ dans le réseau réduit.
- (iii) c^i , le nouveau système de coûts sur le graphe réduit de n_i sommets.
- (iv) J , l'ensemble des sommets du graphe réduit qui sont défendus d'être en deuxième position dans le réseau réduit.
- (v) B , la bijection qui permet de déterminer une tournée optimale dans le réseau de départ à partir d'une tournée optimale du réseau réduit associé au sommet $i = (k_i, \ell_i, \dots)$ de l'arborescence.

Pour la raison de commodité, nous décrivons ces données dans l'ordre qui nous convient, en nous appuyant sur l'exemple de la figure 2 où $n = 7$.

(i) A partir du sommet i de l'arborescence, nous remontons jusqu'à sa racine. Soit K l'ensemble des k_j rencontrés dans ce parcours tels que $\ell_j = \text{oui}$ (k_i inclus, excepté la racine), alors $n_i = n - |K|$. Dans la figure 1, si $i = 7 = (6, \text{non}, \dots)$, $K = \{4, 2\}$ et $n_7 = 7 - 2 = 5$. Le graphe réduit possède donc 5 sommets, le sommet 1 du graphe réduit s'obtient en regroupant les sommets 1, 4 et 2 du graphe initial (voir figure 3 où les nouveaux numéros des sommets du graphe réduit sont écrits à l'extérieur des circonférences contenant les anciens numéros du graphe initial).

(ii) $m_i = m_1 + |K|$. C'est la somme des nombres de personnes qui se trouvent aux sommets de K et m . Dans le cas de notre exemple, $m_7 = m + 2$.

(iii) B est une bijection quelconque de l'ensemble $\{1, 2, \dots, n_i\}$ à l'ensemble $\{1, 2, \dots, n\} - K$ vérifiant $B(n_i) = n$ et $B(1) = k$ où k est le premier des k_j , tels que $\ell_j = \text{oui}$, rencontrés dans le parcours du sommet i de l'arborescence à sa racine. Si $K = \emptyset$, $B(1) = 1$. Dans le cas de notre exemple illustré par la figure 3, on a : $B(1) = 4$, $B(2) = 3$, $B(3) = 5$, $B(4) = 6$ et $B(5) = 7$.

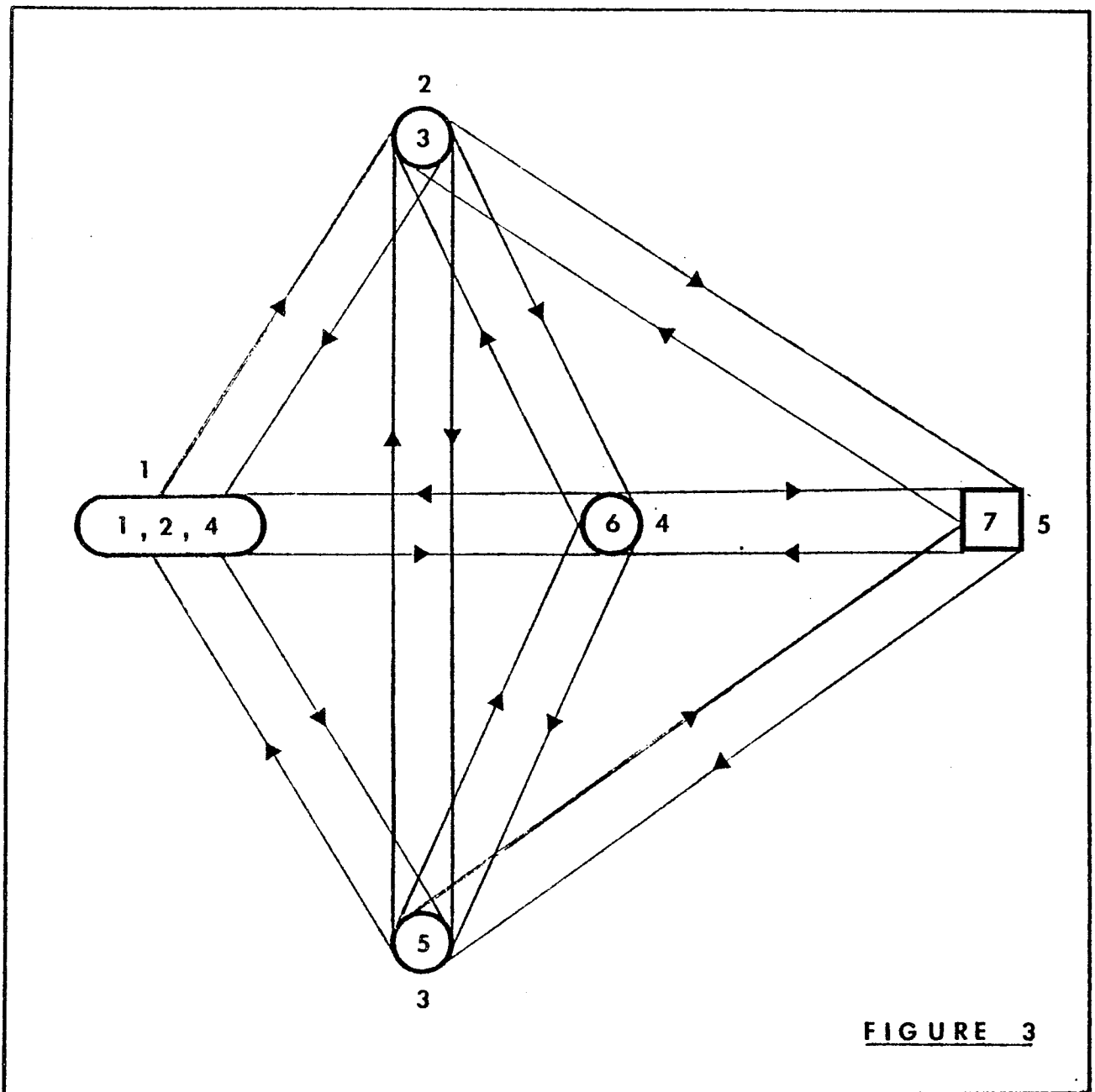


FIGURE 3

(iv) J est l'ensemble des sommets du graphe réduit qui sont défendus d'être en deuxième position dans les tournées associées au sommet i de l'arborescence. Soit $L' = \{\ell_1, \dots, \ell_r\}$ l'ensemble des premières variables logiques consécutives égales à non, à partir du sommet i dans le parcours vers la racine de l'arborescence. Soit $L = \{k_j / \ell_j \in L'\}$, alors $J = \{k / B(k) \in L\}$. Dans le cas de notre exemple, pour $i = 7$, $L' = \{\ell_7\}$, $L = \{6\}$, $J = \{4\}$. Si $\ell_i = \text{oui}$, alors $J = \emptyset$.

(v) c^i est le nouveau système de coûts sur le graphe réduit. Dans cette nouvelle matrice des coûts, les dispositions des valeurs $+\infty$ sont les mêmes que dans la matrice c (voir chapitre I). De plus, on a : $c_{lr}^i = +\infty$ pour $r \in J$ et $c_{rs}^i = c_{B(r)B(s)}$ pour les coûts non égaux à $+\infty$. Dans l'exemple où $i = 7$, $c_{14}^7 = +\infty$, $c_{12}^7 = c_{43}$.

C'est ainsi que se termine la description de l'Etape 3 de la reconstruction des données.

2.5 - Description détaillée de l'Etape 4

Comme dans (i), nous désignons par K l'ensemble des $k_r, k_{r+1}, \dots, k_s$ tels que $\ell_r = \ell_{r+1} = \dots = \ell_s = \text{où}$ dans le parcours du sommet i de l'arborescence à sa racine, rencontrés dans cet ordre. Remarquons que si $\ell_i = \text{où}$, alors $k_r = k_i$. Soit :

$$b_i' = (m+1).c_{1k_s} + (m+2).c_{k_s k_{s-1}} + \dots + (m+s-r+1).c_{k_{r+1} k_r} \quad (2)$$

Dans le cas de notre exemple, on a $K = \{4,2\}$, d'où :

$$b_7' = (m+1).c_{12} + (m+2).c_{24}$$

Supposons d'autre part que $J = \emptyset$, soit b_i'' la valeur de la solution du problème (P') correspondant au réseau réduit caractérisé par n_i, m_i, c^i sur le graphe réduit, il est clair que, dans ce cas, la valeur

$$b_i = b_i' + b_i'' \quad (3)$$

est une borne associée au sommet i de l'arborescence. Si $J = \{2, 3, \dots, n_i - 1\}$, alors $b_i'' = b_i = +\infty$.

Dans les autres cas, seule la valeur de b_i'' change, b_i' reste la même. Pour simplifier les notations, nous supposons que $n_i = n, m_i = m$ et $c^i = c$. On est ainsi conduit à calculer une borne inférieure du coût d'une tournée optimale, dans un réseau décrit au chapitre I, dans lequel on sait qu'aucun sommet de J n'est en deuxième position. Le programme linéaire continu est :

$$(PC2) \quad \begin{cases} y \geq 0 \\ x \geq 0 \\ A1.y = 1 \\ A4.y = 0 \\ -B.y + A2.x = M \\ c.x = z(\min) \end{cases}$$

où les contraintes et la fonction objective sont les mêmes que dans (PC) au chapitre I, sauf $A4.y = 0$ qui résume les contraintes :

$$y_{j2} = 0 \text{ pour } j \in J.$$

Si on pose $A5.y = d$ pour résumer les contraintes $A1.y = \mathbf{1}$ et $A4.y = 0$, on obtient :

$$(PC2) \quad \begin{cases} y \geq 0 \\ x \geq 0 \\ A5.y = d \\ -B.y + A1.x = M \\ c.x = z(\min) \end{cases}$$

En répétant exactement la même décomposition que dans le chapitre II, $A1.y = \mathbf{1}$ étant remplacé par $A5.y = d$, on est conduit à résoudre le problème (P'') défini par :

$$(P'') : \max_{u \in D} w_2(u)$$

où $w_2(u)$ représente la valeur optimale du programme linéaire :

$$(DC2_u) \quad \begin{cases} y \geq 0 \\ A5.y = d \\ u.B.y + u.M = w(u) (\min) \end{cases}$$

Théorème 1 : Soient $u = (u_1, u_2, \dots, u_n, u_{n+1}, u_{n+2}, \dots, u_{n+n})$ et $\sigma = (\sigma_2, \sigma_3, \dots, \sigma_{n-1}) \in \mathfrak{S}_{[2, n-1]}$ tel que :

$$(i) \quad u_{\sigma_2} + u_{n+\sigma_2} = \max_{\substack{2 \leq j \leq n-1 \\ j \notin J}} \{u_j + u_{n+j}\} \text{ et } \sigma_2 \notin J$$

$$(ii) \quad u_{\sigma_3} + u_{n+\sigma_3} \geq u_{\sigma_4} + u_{n+\sigma_4} \geq \dots \geq u_{\sigma_{n-1}} + u_{n+\sigma_{n-1}}, \text{ alors } w_2(u) = \langle u, \theta^\sigma \rangle.$$

Démonstration : C'est la même démonstration que celle du théorème 1 du chapitre II, le lemme 1 étant remplacé par le lemme 2.

Notation 2 : Nous désignons par $\mathfrak{S}_{[2, n-1]}^{(J)}$ l'ensemble des permutations $\sigma = (\sigma_2, \sigma_3, \dots, \sigma_{n-1})$ de $\mathfrak{S}_{[2, n-1]}$ telles que $\sigma_2 \notin J$.

Avec cette notation, on a :

$$w_2(u) = \begin{cases} \min_{\sigma \in \mathfrak{S}_{[2, n-1]}^{(J)}} \langle u, \theta^\sigma \rangle & \text{si } u \in D \\ -\infty & \text{si } u \notin D \end{cases}$$

Il est facile de voir que, pour résoudre (P''), toutes les méthodes proposées dans le chapitre III restent valables puisqu'elles se basent toutes sur la propriété suivante : étant donné un point u , on sait calculer facilement toutes les permutations $\sigma \in \mathfrak{S}_{[2, n-1]}^{(J)}$ telles que $w_2(u) = \langle u, \theta^\sigma \rangle$, et seulement sur cette propriété là.

Donc, même si $J \neq \emptyset$ et $J \neq \{2, 3, \dots, n_i - 1\}$, on est en mesure de calculer b_i'' ; $b_i = b_i' + b_i''$ constitue alors une borne associée au sommet i de l'arborescence.

Remarque 4 : Nous avons vu, à la remarque 2, que le principe de la première méthode de partitionnement est de chercher à descendre dans la bonne branche de l'arborescence. Pour la deuxième méthode, nous disons qu'elle cherche surtout à augmenter les valeurs des bornes associées aux sommets de l'arborescence et, par conséquent, d'en éliminer au maximum. En effet, considérons par exemple l'ensemble R_0 de toutes les tournées à la racine de l'arborescence. Selon ce principe de partitionnement, si u^* est une solution du problème (P') calculée par l'une des méthodes du chapitre III, un sommet q est choisi pour séparer R_0 en R_{01} (l'ensemble des tournées qui n'admettent pas q en deuxième position) et R_{02} (l'ensemble des tournées qui admettent q en deuxième position) si on a (Remarque 1) :

$$2 \leq q \leq n-1 \quad \text{et} \quad c_{1q} - u_1^* - u_{n+q}^* = 0 \quad (4)$$

Or, d'après la reconstruction des données, dans le calcul de la borne associée à R_{01} , c_{1q} est porté égal à $+\infty$. Si q est le seul entier compris entre 2 et $n-1$ tel que la relation (4) soit vérifiée, ce qui arrive souvent dans la pratique, le fait de porter c_{1q} égal à $+\infty$ implique que l'on peut augmenter la valeur de u_1^* et donc la valeur de la borne associée à R_{01} .

2.6 - Description détaillée de l'Etape 5

Définition 3 : Dans ces procédures de SEP, nous disons qu'un sommet $i = (k_i, \ell_i, b_i, f_i)$ de l'arborescence est terminal s'il existe une tournée T dans le réseau réduit associé à i telle que $b_i'' = c_i^i(T)$.

Considérons un sommet terminal $i = (k_i, \ell_i, b_i, p_i)$ de l'arborescence.

Soit $T = (1, t_2, t_3, \dots, t_{n_i-1}, n_i)$ une tournée de R_i telle que $b_i'' = c^i(T)$.

Soit K l'ensemble des $k_r, k_{r+1}, \dots, k_s$ tels que $\ell_r = \ell_{r+1} = \dots = \ell_s = \text{oui}$, rencontrés dans cet ordre, dans le parcours du sommet i de l'arborescence à sa racine. Soit T^* la tournée du réseau initial définie par :

$$T^* = (1, k_s, k_{s-1}, \dots, k_r, B(t_2), B(t_3), \dots, B(t_{n_i-1}), n)$$

Il est facile de voir que $c(T^*) = b_i' + b_i'' = b_i$.

D'après la proposition 1, si le sommet terminal i est choisi comme noeud de séparation, alors T^* est une tournée optimale dans le réseau initial.

Dans notre exemple, toujours pour $i = 7$, $K = \{4, 2\}$, supposons que $T = (1, 3, 2, 4, 5)$ est telle que $c^7(T) = b_7''$ et que le sommet 7 est choisi comme noeud de séparation, alors $T^* = (1, 2, 4, 5, 3, 6, 7)$ est une tournée optimale.

2.7 - Algorithme de SEP

Avant de donner l'algorithme, nous précisons que :

(i) A chaque fois que nous faisons intervenir l'Etape 2, c'est la deuxième méthode de partitionnement qui est mise en oeuvre.

(ii) A chaque fois que nous faisons intervenir l'Etape 4, c'est la méthode de "sous-gradient généralisée" qui est mise en oeuvre pour calculer la valeur de b_i'' et ensuite la valeur de la borne b_i associée au sommet i de l'arborescence.

Algorithme

- Pas (0) - Poser $i := 0$, $j := 0$; calculer b_0 comme Etape 4 et p_0 comme Etape 2 ;
- Pas (1) - Faire $i := i+1$, $k_i = p_j$, $\ell_i := \text{non}$; reconstruire les données comme Etape 3 ; calculer b_i comme Etape 4 et p_i comme Etape 2 ; faire $i := i+1$, $k_i = p_j$, $\ell_i := \text{oui}$; reconstruire les données comme Etape 3 ; calculer b_i comme Etape 4 et p_i comme Etape 2 ;
- Pas (2) - Choisir, parmi les sommets pendants de l'arborescence, un sommet j comme Etape 1. Si ce sommet j est terminal, aller en (3) sinon aller en (1) ;
- Pas (3) - Soit T une tournée dans le réseau réduit relatif au sommet j telle que $c^j(T) = b_j''$. Déterminer une tournée optimale T^* comme Etape 5.

Une version de cet algorithme a été programmée en algol W et essayée sur l'ordinateur IRIS 80. Dans ces expériences, les coûts c_{ij} sont des nombres compris entre 0 et 100, gérés aléatoirement. La courbe de la figure 4 représente la variation du temps de calcul (en minutes) en fonction de la taille n du problème.

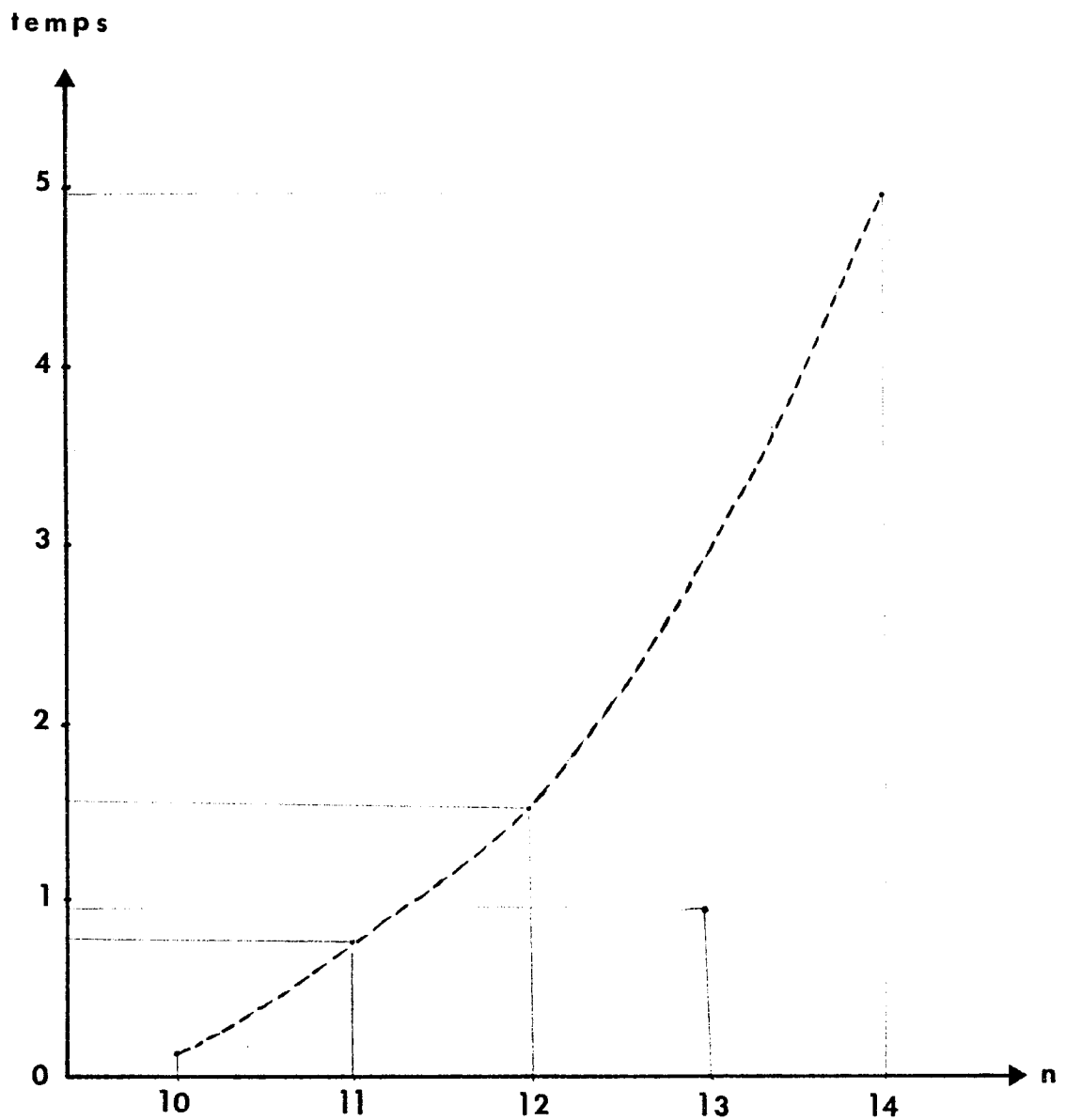


FIGURE 4

Le tableau 1 donne les résultats complémentaires concernant les coûts des tournées optimales, les valeurs des premières bornes à la racine de l'arborescence et les nombres des sommets de l'arborescence explorés.

n : le nombre de sommets du graphe	10	11	12	13	14
m : m+1 personnes partent du sommet 1	4	0	5	6	6
Nombre de sommets de l'arborescence	37	193	191	123	459
Première borne à la racine de l'arborescence	1136	588	960	1076	1220
Coûts de la tournée optimale trouvée	1246	696	1224	1333	1735

TABLEAU 1

§3 - DEUXIEME ALGORITHME DE SEP

Il ne s'agit pas de donner ici un algorithme tout à fait différent de celui qui a été proposé dans le paragraphe précédent, il s'agit tout simplement de le modifier un peu afin de pouvoir résoudre un problème de taille plus grande. En effet, les résultats numériques présentés ci-dessus ont montré que les premières bornes calculées à la racine de l'arborescence ne sont pas assez proches des coûts des tournées optimales, alors que le calcul de ces bornes prend un temps non négligeable. Ces remarques nous conduisent à rechercher une autre façon d'approcher le coût d'une tournée optimale, très peu coûteuse en temps de calcul et telle que la borne obtenue ne soit pas trop mauvaise.

Après plusieurs expériences, nous constatons que les valeurs fournies par l'initialisation (2) du chapitre III ne sont pas trop éloignées des solutions optimales données par la résolution du problème (P'). Nous prenons donc comme borne la valeur correspondant à cette initialisation et nous modifions ainsi l'Etape 4 de l'algorithme précédent, et seulement l'Etape 4. Les résultats numériques, essayés d'une part sur les mêmes exemples précédents et d'autre part sur les exemples de plus grande taille dont les coûts sont toujours gérés aléatoirement entre 0 et 100, sont résumés de la même façon par la courbe de la figure 5 et par le tableau 2.

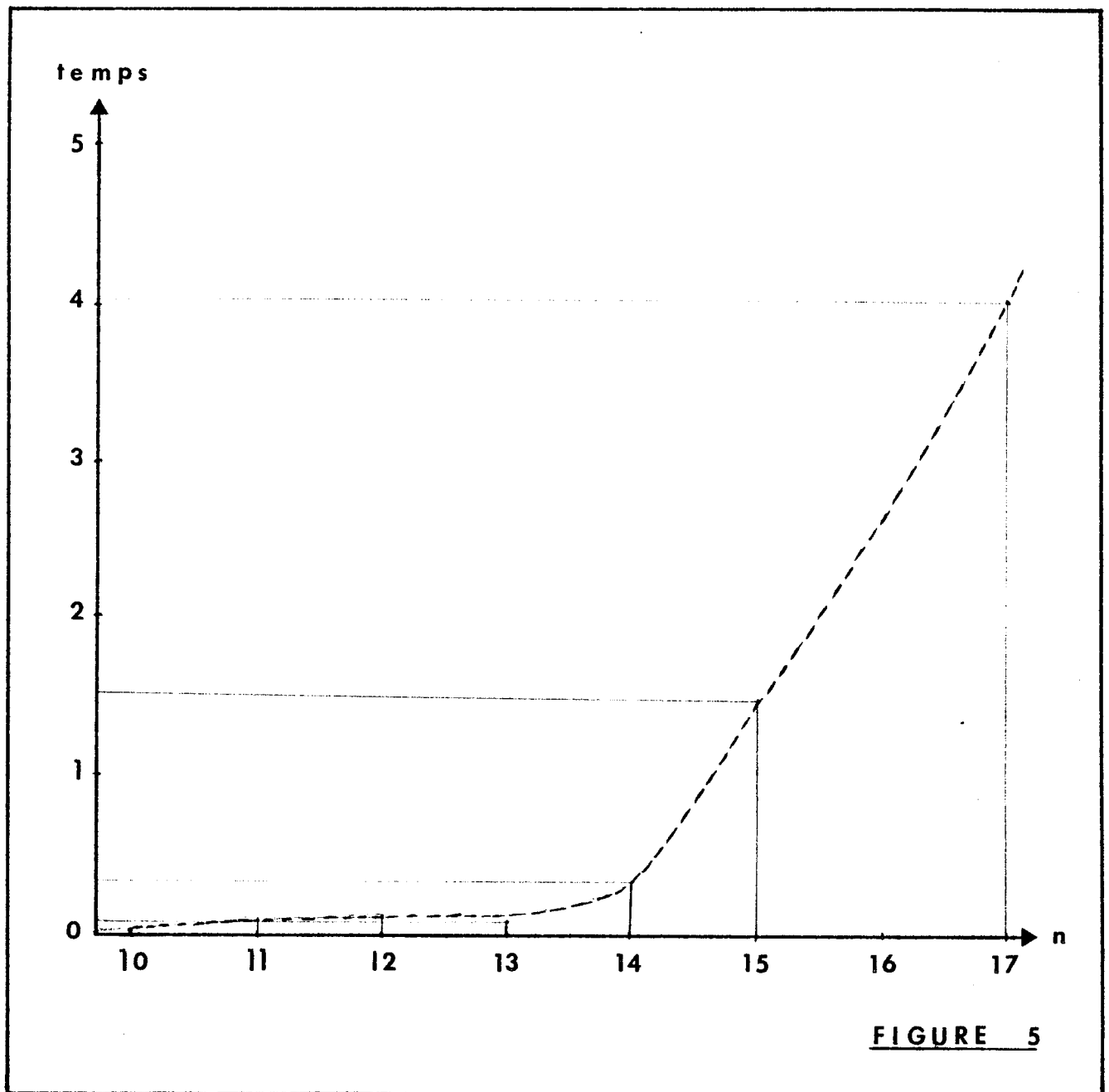


FIGURE 5

n : nombre de sommets du graphe	10	11	12	13	14	15	17
m : (m+1) personnes partent du sommet 1	4	0	5	6	6	6	5
Nombre de sommets de l'arborescence explorés	41	199	239	169	739	2095	5057
Première borne à la racine de l'arborescence	1034	544	928	1009	1115	1111	1088
Coût de la tournée optimale	1246	696	1224	1333	1735	2345	1942

TABLEAU 2

§4 - VARIATION DU PARAMETRE m

Nous avons vu au paragraphe 2 du chapitre I, qu'il existe une valeur $m_0 \in \mathbb{N}$ telle que pour $m \geq m_0$, la solution du problème (P) est celle du problème du voyageur de commerce. Cette propriété permet au paramètre m de jouer le rôle de pondération entre les deux critères (iv)-a et (iv)-b présentés dans l'introduction de l'exposé. Elle permet en outre de nous rendre compte de la difficulté du problème (P) qui est au moins égale à celle du problème du voyageur de commerce, tant sur le plan théorique que sur le plan pratique. Les résultats suivants montrent que, sur le plan pratique, le problème (P) est bien plus difficile que le problème du voyageur de commerce. Pour mieux l'exposer, il est nécessaire de nous rappeler que le problème (P) dépend de trois paramètres : le nombre n de sommets du graphe, le système de coûts c sur ses arcs et le paramètre m dont la signification est donnée au chapitre I. Nous supposons dans la suite, sauf indication contraire, que n et c restent constants et nous notons respectivement par $P(m)$ et $P'(m)$ les problèmes (P) et (P') correspondant à la valeur de m . Nous appelons tout simplement chemins hamiltoniens les chemins hamiltoniens de 1 à n .

Proposition 2 : Soient T_0 une tournée optimale du problème $P(0)$ et H_0 un chemin hamiltonien optimal. Si T_0 et H_0 désignent le même chemin hamiltonien, alors quelque soit m , T_0 reste une tournée optimale du problème $P(m)$.

Démonstration : Rappelons nous d'abord que, si T désigne une tournée quelconque dans le problème $P(m)$, alors :

$$c(T) = c'(T) + m.c(H)$$

où $c'(T)$ est le coût de la tournée T dans le problème $P(0)$ et où H désigne le même chemin hamiltonien que T (voir démonstration du théorème 1 du chapitre I).

Prenons $m \neq 0$ et supposons que T_0 n'est pas une solution optimale du problème $P(m)$, alors il existe une tournée T_1 telle que :

$$c(T_1) < c(T_0)$$

$$\text{D'où } c'(T_1) + m.c(H_1) < c'(T_0) + m.c(H_0)$$

où H_1 désigne le même chemin hamiltonien que T_1 .

Donc, d'après l'optimalité de T_0 dans le problème $P(0)$, on a :

$$0 \leq c'(T_1) - c'(T_0) < m.(c(H_0) - c(H_1))$$

ou $c(H_1) < c(H_0)$

ce qui est contraire à l'optimalité de H_0 . $\square$

Proposition 3 : Soient T_0 une tournée optimale du problème $P(m)$ et H_0 un chemin hamiltonien optimal. Si T_0 et H_0 désignent le même chemin hamiltonien, alors quel que soit $m' > m$, T_0 reste une tournée optimale du problème $P(m')$.

Démonstration : La même que pour la proposition 2.

Corollaire 1 : Il existe une valeur $m_0 \in \mathbb{N}$ telle que, quelque soit $m \geq m_0$, la solution du problème $P(m)$ reste la même.

Démonstration : Ce corollaire est conséquence directe du théorème 1, chapitre et de la proposition 3. $\square$

Donc à partir d'une certaine valeur de m , la solution du problème $P(m)$ ne change plus. Il n'en est pas de même pour les opérations qui permettent de déterminer cette solution, en particulier le nombre de sommets de l'arborescence explorés, les bornes calculées pour chacun de ces sommets, les temps de calcul, etc... . Cependant, il nous semble difficile de choisir un critère précis pour étudier ces variations en fonction de m . Il est naturel de penser que plus le rapport $b/c(T_0)$ entre la borne calculée et le coût d'une tournée optimale est proche de l'unité, plus court est le temps de calcul pour déterminer cette tournée. Ceci n'est pas tout à fait vrai. En effet, le coût c étant donné, il est facile de voir que les opérations faites pour déterminer une tournée optimale ainsi que cette tournée optimale elle-même ne changent pas si l'on remplace c par c^1 , obtenu en ajoutant à tous les c_{ij} une même constante k . Par contre, le rapport entre la borne calculée et le coût d'une tournée optimale sera égale à :

$$\frac{b + (2m+n)(n-1) \cdot k/2}{c(T_0) + (2m+n)(n-1) \cdot k/2} ,$$

quantité qui tend vers l'unité quand k tend vers l'infini.

On peut également penser que la différence entre le coût d'une tournée optimale et la borne calculée donne une idée sur ces variations. Ceci n'est pas tout à fait vrai, non plus. En effet, il est facile de voir que, si l'on multiplie tous les c_{ij} par une constante k , les opérations faites pour déterminer une tournée optimale ainsi que cette tournée optimale elle-même ne changent pas. Par contre, la différence $c(T_0) - b$ devient $k(c(T_0) - b)$, quantité qui tend vers l'infini quand k tend vers l'infini (sauf si $c(T_0) - b = 0$).

Les deux raisons ci-dessus montrent que, pour définir un critère de comparaison, il est préférable de tenir compte aussi de la valeur moyenne des coûts. C'est ainsi que nous sommes décidés de prendre comme critère, pour comparer le problème $P(o)$ et le problème du voyageur de commerce qui correspond à $P(m)$ pour m assez grand, la différence entre le coût d'une tournée optimale et la valeur de la solution donnée par l'initialisation (2) du chapitre III. Ici, ce critère est indépendant de la valeur moyenne des coûts puisqu'il s'agit du même réseau.

Soit d_o cette différence correspondant au problème $P(o)$. D'autre part, d'après la remarque 5 du chapitre II, le programme linéaire (PC'), obtenu en oubliant les contraintes d'intégrité du programme (PLE') est équivalent au programme (PC) dans lequel m "tend vers" l'infini. Ce programme (PC') est la formulation utilisée dans [17] pour résoudre le problème du voyageur de commerce. La même initialisation (2) du chapitre III donne aussi une solution initiale du programme (DC'), dual de (PC'). Soit d_1 la différence entre le coût d'un chemin hamiltonien optimal et la borne donnée par cette solution initiale, on a :

Proposition 4 : $d_o \geq d_1$.

Démonstration : Soit u^o la solution initiale commune aux deux problèmes. Sans perdre de généralité, supposons que $T_o = (1, 2, \dots, n)$ est une tournée optimale et soit H_o un chemin hamiltonien optimal. On a :

$$\begin{aligned} d_o &= c(T_o) - \langle u^o, \theta^o \rangle \quad \text{avec } \theta^o \in \mathcal{G}(u^o) \\ &\geq (c_{12} - u_1^o - u_{n+2}^o) + 2(c_{23} - u_2^o - u_{n+3}^o) + \dots + \\ &\quad (n-1)(c_{(n-1)n} - u_{n-1}^o - u_{n+n}^o). \end{aligned}$$

D'autre part, d'après la forme de (DC'), on a :

$$\begin{aligned} d_1 &= c(H_o) - (u_1^o + u_2^o + u_{n+2}^o + \dots + u_{n-1}^o + u_{n+n-1}^o + u_{n+n}^o) \\ &\leq (c_{12} - u_1^o - u_{n+2}^o) + (c_{23} - u_2^o - u_{n+3}^o) + \dots + \\ &\quad (c_{(n-1)n} - u_{n-1}^o - u_{n+n}^o) \\ &\leq d_o \quad \square \end{aligned}$$

Ce résultat montre que, selon ce critère de comparaison, pour deux formulations équivalentes, l'une pour le problème de ramassage scolaire, l'autre pour le problème du voyageur de commerce, le premier problème est plus difficile que le second. Une expérience faite sur un exemple où $n = 12$, où c désigne les distances entre les points gérés aléatoirement dans un carré de 50 unités de côté et où la méthode de séparation est la première de l'Etape 2, donne les variations du nombre de sommets de l'arborescence explorés, du temps de calcul en minutes (sur IRIS 80), de la borne calculée et du coût de la tournée optimale en fonction de m .

Valeur de m	30	200	10^6
Nombre de sommets de l'arborescence explorés	2501	1721	1711
Temps de calcul (en minutes)	1,60	1,04	1,02
Initialisation	2800	16230	$7900034 \cdot 10^7$
Coût de la tournée optimale	4195	23915	$11600006 \cdot 10^8$

TABLEAU 3

CONCLUSION

Parmi les méthodes de résolution des problèmes NP-complets, les procédures de SEP demeurent les plus fréquemment utilisées. L'efficacité de ces procédures varie beaucoup avec la qualité de l'évaluation proposée.

En ce qui concerne notre problème (P) de recherche de tournée optimale, cette évaluation se fait par la résolution du programme linéaire (PC), de grandes dimensions. Malgré la performance des méthodes de résolution de ce programme linéaire qui dépasse largement celle du *simplexe* classique, les procédures de SEP que nous avons utilisées pour résoudre le problème (PC) restent impuissantes : 4,97 minutes de calcul sur l'IRIS 80 pour un réseau de 14 sommets. En remplaçant la valeur de la solution optimale du programme linéaire par celle qui correspond à une solution initiale du programme dual, le résultat obtenu s'est sensiblement amélioré : 4,04 minutes pour un réseau de 17 sommets. Ceci vient du fait que la borne donnée par le programme (PC) reste assez éloignée du coût d'une tournée optimale : 680 par rapport à 824 sur un réseau de 9 sommets, 793 par rapport à 1124 sur un autre réseau de même nombre de sommets... . Cet écart relativement grand entre la borne trouvée et le coût d'une tournée optimale s'explique facilement par la formation de circuits dans le graphe partiel, obtenu à partir du graphe initial en effaçant les arcs (i,j) sur lesquels les variables x_{ij} sont nulles à l'optimum du programme linéaire (PC). Ce phénomène est d'ailleurs bien connu dans la formulation du problème du voyageur de commerce comme problème d'affectation [17]. Parmi ces circuits, ceux de longueur 2 (de forme (i,j,i)) jouent un rôle capital. Reprenons l'exemple de la remarque 7 du chapitre I, l'existence du circuit $(2,3,2)$ implique que l'écart entre la borne calculée et le coût de la tournée optimale est supérieur ou égal à 2. Beaucoup d'autres exemples numériques confirment ce phénomène, surtout dans le cas où le système de coûts est symétrique, par exemple les distances euclidiennes entre les points aléatoires qui font partie de la troisième série de l'essai. En effet, dans ce cas c_{ij} est égal à c_{ji} ; si l'on suppose d'autre part que c_{ij} est petit par rapport aux autres coûts alors, à l'optimum du programme (PC), les variables de trafics x_{ij} et x_{ji} ont une très grande chance d'être non nulles. Le cas des distances euclidiennes entre les points aléatoires constitue aussi un exemple difficile pour les procédures de SEP puisqu'on y trouverait beaucoup de tournées non optimales dont les coûts diffèrent très peu de leur minimum et que, par conséquent, il est difficile de les éliminer de l'arborescence d'exploration. Cette situation est d'ailleurs confirmée par les expériences pratiques.

Pour supprimer ces inconvénients, il nous semble intéressant d'enlever les circuits de longueur 2 du graphe partiel correspondant à une solution optimale du programme (PC). Plus précisément, soient u^1 une solution optimale du problème (P') et $J^1 = \{(i,j)/u_i^1 + u_{n+j}^1 = c_{ij}\}$. Soit G^1 le graphe partiel obtenu à partir de G en supprimant les arcs (i,j) n'appartenant pas à J^1 . Pour tout circuit (i,j,i) du graphe G^1 , nous associons deux contraintes supplémentaires :

$$x_{ij} + x_{ji} \leq \sum_{k=1}^n x_{ik} \quad (1)$$

$$x_{ij} + x_{ji} \leq \sum_{k=1}^n x_{jk} \quad (2)$$

qui s'ajoutent aux contraintes de (PC) et on obtient ainsi un nouveau programme linéaire (PC1). La même méthode de décomposition s'applique à (PC1) et conduit à résoudre le problème (P1) qui, en plus des variables u , possède des variables supplémentaires correspondant aux contraintes du type (1) et (2) que nous venons d'ajouter. On peut également envisager d'enlever les circuits de longueurs 3, 4, Il ne s'agit là que des premières réflexions sur cette façon de procéder et la démarche reste ouverte.

Reste ouvert également le problème de recherche d'une formulation qui enlève tous les circuits du graphe partiel construit sur les arcs de trafics non nuls, en généralisant la formulation utilisée dans [5] et [6] pour résoudre le problème du voyageur de commerce. Une telle formulation est possible et serait assez efficace pour les procédures SEP. Par contre, le programme linéaire obtenu est beaucoup plus difficile à résoudre que celui de [5] et [6].

L'intérêt principal de notre travail réside dans les quatre méthodes de résolution du programme linéaire (PC), surtout dans la dernière d'entre elles qui exploite à la fois l'avantage de l'unimodularité de la matrice $A2$ et celui des méthodes de sous-gradient. Elle nous permet d'éviter les erreurs de précision rencontrées dans la première méthode et se généralise facilement pour résoudre les problèmes du type :

$$\left\{ \begin{array}{l} y \geq 0, \quad x \geq 0 \\ A1.y = a^1 \\ B.y + A2.x = a^2 \\ c^1.y + c^2.x = z(\min) \end{array} \right.$$

où $A2$ est une matrice unimodulaire et où le programme

$$\left\{ \begin{array}{l} y \geq 0 \\ A^1.y = a^1 \\ (c^1 - u.B).y = z(u) \text{ (min)} \end{array} \right.$$

est un problème qui se résoud rapidement au point de vue de temps de calcul.

BIBLIOGRAPHIE

- [1] BERGE
"Graphes et hypergraphes". Dunod 1970.
- [2] M. BELLMORE and G.L. NEMHAUSER
"The traveling-Salesman problem : A survey", Operations Research
16(1968) 538-558.
- [3] BENDERS
"Partitionning procedures for solving mixed variables programming problems". Num. Math., 4(1962) 238-252.
- [4] P.H. FARGIER
"Une méthode de détermination d'un réseau de transport optimal".
Thèse de Docteur-Ingénieur, U.S.M.G. (1977).
- [5] M. HELD and R.M. KARP
"The Traveling-Salesman problem and minimum spanning tree".
Operations Research 18(1970) 1138-1162.
- [6] M. HELD and R.M. KARP
"The Travelong-Salesman problem and minimum spanning tree".
Math. Programming, 1(1971) 6-25.
- [7] M. HELD, P. WOLFE and H.P. CROWDER
"Validation of subgradient optimization". Math. Programming, 6(1974)
62-88.
- [8] R.M. KARP
"Reductibility among combinatorial problems". R.E. Miller and J.W. Thatcher
(eds) (1972) 85-104.
- [9] P.J. LAURENT
"Approximation et Optimisation". Hermann, 1971.
- [10] J.D.C. LITTLE, K.G. MURTY, D. SWEENEY, C. KAREL
"An algorithm for the Traveling-Salesman problem". Operations Research
11(1963) 972-990.
- [11] C. MUKENDI
"Sur l'implantation d'équipement dans un réseau : le problème de m-
centre". Thèse de 3ème cycle, U.S.M.G. (1975).
- [12] T.S. MORZKIN and I.J. SCHOENBERG
"The relaxation method for linear inequalities". Canadian Journal
of mathematics, 6(1954) 393-404.
- [13] B.T. POLJAK
"A general method of solving extremum problems". Soviet mathematics
Doklady 8(1967) 593-597.
- [14] M. SAKAROVITCH
"Programmation linéaire". Cours photocopié, IRMA, (1973).

- [15] M. SAKAROVITCH
"Introduction à l'étude des graphes". Cours photocopié, IRMA (1975).
- [16] M. SAKAROVITCH
"Introduction à l'optimisation combinatoire". Cours photocopié, IRMA,
à paraître.
- [17] D. SHAPIRO
"Algorithms for the solution of the Optimal Cost Traveling-Sales-
man problem". Sc. D. Thesis, Washington University, St Louis (1966).
- [18] M. SIMONNARD
"Programmation linéaire". 39 Finance et Economie Appliquée. Dunod (1972)
- [19] J.P. UHRY
"Méthode de sous-gradient dans la décomposition de programmes linéaires"
Mimeo.

RAPPEL DES NOTATIONS

$ K $	Cardinalité de l'ensemble K .
$A-B$	Ensemble des éléments de l'ensemble A n'appartenant pas à l'ensemble B .
$\mathcal{S}_{[2,n-1]}$	Ensemble des permutations de l'ensemble $\{2,3,\dots,n-1\}$.
$ v $	Norme euclidienne du vecteur v .
$A.B$	Produit des deux matrices A et B .
$s.t$	Produit de deux nombres s et t .
$\langle u, \theta \rangle$	Produit scalaire des vecteurs u et θ .
$P_D(v)$	Projection du point v sur le domaine D .
$\partial f(x)$	Sous-différentiel de la fonction f au point x .
$ t $	Valeur absolue du nombre réel t .
$\overset{\circ}{Q}$	Intérieur de l'ensemble Q

Dernière page d'une thèse

VU

Grenoble, le 26 octobre 1978

Le Président de la thèse

A. S. Kharvitch
A. S. Kharvitch

Vu, et permis d'imprimer,

Grenoble, le 26 octobre 1978

Le Président de l'Université
Scientifique et Médicale



J. G. Cau
D. G. CAU