

Polycopié de FMA_0EL01_TP

Optimisation continue et analyse numérique¹

Irène Charon

Olivier Hudry

18 septembre 2024

1. Les chapitres de ce polycopié consacrés à l’optimisation sont extraits, avec des adaptations, du livre « Introduction à l’optimisation continue et discrète », par Irène Charon et Olivier Hudry, Lavoisier, 2019.

Table des matières

I	Éléments d’analyse numérique	7
1	Analyse matricielle - Généralités	9
1.1	Rappels d’algèbre linéaire	9
1.1.1	Adjointes	9
1.1.2	Types de matrices	10
1.1.3	Spectre d’une matrice	10
1.1.4	Réduction d’une matrice	10
1.1.5	Valeurs singulières	11
1.2	Normes	12
2	Problèmes de l’analyse numérique	15
2.1	Erreurs	15
2.2	Conditionnement d’un système linéaire	16
3	Résolution de systèmes linéaires	21
3.1	Généralités	21
3.2	Méthode de Gauss	22
3.2.1	Étape d’élimination	22
3.2.2	Choix du pivot	24
3.2.3	Complexité	25
3.2.4	Variante : la méthode de Gauss-Jordan	25
3.3	Factorisation LU	27
3.4	Méthode de Cholesky	30
4	Valeurs et vecteurs propres	33
4.1	Méthode de Jacobi	34

II	Optimisation linéaire	41
5	Optimisation linéaire : l’algorithme du simplexe	43
5.1	Introduction	43
5.2	L’algorithme du simplexe sur un exemple	47
5.3	Définitions et terminologie	51
5.4	Résumé d’une itération	53
5.5	La dégénérescence et le cyclage	54
5.6	Recherche d’un dictionnaire réalisable	58
5.7	Exercices	61
6	Dualité en optimisation linéaire	73
6.1	Définition du problème dual	73
6.2	Théorème de la dualité	74
6.3	Le théorème des écarts complémentaires : un certificat d’opti- malité	78
6.4	La signification économique du dual	80
6.5	Problème dual-réalisable	83
6.6	Exercices	84
7	Flot de valeur maximum et coupe de capacité minimum	93
7.1	Introduction	93
7.2	Définitions, notations et problèmes	94
	Problème du flot de valeur maximum comme problème d’optimisation linéaire	96
7.3	Résultats théoriques	96
7.4	Algorithme de Ford et Fulkerson	98
	7.4.1 Chaîne augmentante	98
	7.4.2 Description de l’algorithme de Ford et Fulkerson	101
	7.4.3 Preuve, convergence et complexité de l’algorithme . . .	104
	7.4.4 Exemple d’application de l’algorithme de Ford et Ful- kerson	107
7.5	Application à un problème d’affectation	110
	7.5.1 Un exemple et quelques définitions	110
	7.5.2 Modélisation d’un problème d’affectation	110
	7.5.3 Traitement de l’exemple de la partie 7.5.1	112
7.6	Application à un problème de transport	114
7.7	Exercices	119

TABLE DES MATIÈRES

5

III Optimisation continue non linéaire 135

8 Optimisation non linéaire sans contrainte 137

8.1 Introduction	137
8.2 Optimisation unidimensionnelle	138
8.2.1 Méthode de Newton	138
8.2.2 Dichotomie pour une fonction dérivable	139
8.2.3 Interpolation quadratique	141
8.2.4 Dichotomie sans dérivation pour une fonction unimodale	141
8.3 Généralités pour l’optimisation multidimensionnelle	142
8.3.1 Notions de topologie	143
8.3.2 Gradient	144
8.3.3 Matrice hessienne	145
8.4 Condition nécessaire et condition suffisante d’optimalité locale	145
8.5 Fonctions convexes	148
8.6 Fonctions quadratiques	149
8.7 Méthodes de descente	150
8.7.1 Généralités	151
8.7.2 Vitesse de convergence	151
8.7.3 Méthodes de gradient	152
8.7.4 Méthode de la plus forte pente à pas fixe	153
8.7.5 Méthode de la plus forte pente à pas optimal	153
8.7.6 Méthode de la plus forte pente accélérée	155
8.8 Méthode de Newton	156
8.9 Exercice	159

9 Optimisation non linéaire avec contraintes 163

9.1 Généralités	163
9.2 Conditions de Lagrange	171
9.3 Conditions de Karush, Kuhn et Tucker	172
9.4 Méthodes de descente	177
9.5 Cas des fonctions convexes	180
9.5.1 Généralités	180
9.5.2 Linéarisation : introduction	182
9.5.3 Linéarisation : méthode de Frank et Wolfe	184
9.6 Exercices	191

A	Rappels sur les graphes	201
A.1	Graphes non orientés	201
A.1.1	Notions fondamentales	201
A.1.2	Chaînes et cycles	203
A.1.3	Connexité	204
A.1.4	Graphes particuliers	204
A.2	Graphes orientés	206
A.2.1	Notions fondamentales	206
A.2.2	Chemins et circuits	211
A.2.3	Forte connexité	212
	Bibliographie	215
	Index	218

Première partie

Éléments d'analyse numérique

Chapitre 1

Analyse matricielle - Généralités

1.1 Rappels d’algèbre linéaire

1.1.1 Adjoint

Dans toute la suite, on considère \mathbb{R} ou \mathbb{C} comme corps de base. Rappelons d’abord quelques définitions.

Étant donné un vecteur x (représenté généralement dans ce polycopié par une matrice colonne), on appelle *adjoint* de x et on note x^* le vecteur trans-

posé du vecteur conjugué de x : si $x = \begin{pmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{pmatrix}$, alors $x^* = (\overline{x_1}, \dots, \overline{x_i}, \dots, \overline{x_n})$.

Remarque

Si on se place dans \mathbb{R} , on a : $x^* = (x_1, \dots, x_i, \dots, x_n)$.

Le *produit hermitien* de deux vecteurs x et y de dimension n est défini par : $(x, y) = \sum_{i=1}^n \overline{x_i} y_i$. Si les vecteurs sont représentés par des vecteurs colonnes, on a : $(x, y) = x^* y$, où le produit est le produit matriciel. Si les vecteurs sont à composantes réelles, le produit hermitien devient le *produit scalaire euclidien* : $(x, y) = \sum_{i=1}^n x_i y_i = x^t y$.

À une matrice A , on peut associer sa *matrice adjointe*, notée A^* . Celle-ci est définie comme suit : si $A = (a_{i,j})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}}$, alors $A^* = \overline{A^t} = (\overline{a_{j,i}})_{\substack{1 \leq j \leq p \\ 1 \leq i \leq n}}$.

On a : $(A^*)^* = A$.

Si x et y sont deux vecteurs colonnes ayant respectivement n lignes et p lignes et A une matrice à n lignes et p colonnes, on peut vérifier la propriété : $(x, Ay) = (A^*x, y)$.

1.1.2 Types de matrices

Une matrice carrée réelle A est dite :

- *symétrique* si $A^t = A$,
- *normale* si $AA^t = A^tA$,
- *orthogonale* si $AA^t = A^tA = I$, où I désigne la matrice identité.

Dans la suite, les qualificatifs « symétrique » et « orthogonale » ne s'appliquent qu'à des matrices réelles.

Une matrice carrée complexe A est dite :

- *hermitienne* si $A^* = A$,
- *normale* si $AA^* = A^*A$.
- *unitaire* si $A^*A = AA^* = I$.

On remarquera qu'une matrice symétrique ou hermitienne est normale. De même pour une matrice orthogonale ou unitaire. On rappelle que les valeurs propres d'une matrice réelle symétrique ou d'une matrice hermitienne sont réelles.

1.1.3 Spectre d'une matrice

Une *valeur propre* d'une matrice carrée A est un scalaire λ tel qu'il existe un vecteur x non nul vérifiant : $Ax = \lambda x$. Le vecteur x est alors dit *vecteur propre* de A .

Soit A une matrice carrée. Le *spectre* de A est l'ensemble des valeurs propres de A . Le *rayon spectral* de A est le plus grand des modules des valeurs propres de A ; il est noté $\rho(A)$.

1.1.4 Réduction d'une matrice

Deux matrices carrées sont dites *semblables* si elles sont susceptibles de représenter la même application linéaire sur deux bases différentes. Si A et B sont deux matrices semblables, il existe une matrice inversible P vérifiant $A = P^{-1}BP$; la matrice P s'appelle *matrice de passage*. Une matrice est *diagonalisable* si elle est semblable à une matrice diagonale; cette matrice

1.1. RAPPELS D’ALGÈBRE LINÉAIRE

11

diagonale est constituée des valeurs propres de A comptées avec leur ordre de multiplicité.

Une matrice symétrique réelle est semblable à une matrice diagonale réelle.

Une matrice carrée est inversible si et seulement si elle ne possède aucune valeur propre nulle.

Une matrice symétrique réelle est semblable à une matrice diagonale réelle.

En fait on peut aussi démontrer, pour les matrices carrées, les résultats suivants :

Théorème 1.1.

1. Soit A une matrice carrée quelconque ; il existe une matrice unitaire U telle que $U^{-1}AU$ soit triangulaire.
2. Soit A une matrice normale ; il existe une matrice unitaire U telle que $U^{-1}AU$ soit diagonale.
3. Soit A une matrice symétrique ; il existe une matrice orthogonale O telle que $O^{-1}AO$ soit diagonale.

Corollaires des définitions et de ce théorème :

1. Les modules des valeurs propres d’une matrice orthogonale ou unitaire valent 1.
2. Une matrice hermitienne (resp. symétrique) ou unitaire est diagonalisable par une matrice de passage unitaire (resp. orthogonale).
3. Une matrice orthogonale O est diagonalisable par une matrice U , en général non réelle, unitaire ($O = U^*DU$), les éléments diagonaux de D étant de module 1.

1.1.5 Valeurs singulières

La matrice A^*A est normale, elle est donc diagonalisable. On peut montrer facilement que ses valeurs propres sont positives ou nulles. On appelle *valeurs singulières* de A les racines carrées positives des valeurs propres de A^*A . La matrice A est inversible si et seulement si ses valeurs singulières sont toutes strictement positives.

Deux matrices A et B sont dites *équivalentes* s’il existe deux matrices inversibles U et V telles que $B = U^{-1}AV$.

Soit A une matrice carrée ; A est équivalente à une matrice diagonale dont la diagonale est constituée des valeurs singulières de A . Plus précisément :

- si A est réelle, il existe deux matrices carrées orthogonales U et V et une matrice diagonale D constituée des valeurs singulières de A telles que : $A = U^t D V$;
- si A est complexe, il existe deux matrices carrées unitaires U et V et une matrice diagonale D constituée des valeurs singulières de A telles que : $A = U^* D V$.

1.2 Normes

Nous aurons besoin dans ces chapitres non seulement de la notion de norme vectorielle mais également de norme matricielle : nous allons donc rappeler quelques notions concernant les premières et définir les secondes. Soit $x = (x_i)_{1 \leq i \leq n}$ un vecteur. Les trois normes vectorielles les plus usuelles sont les suivantes :

- $\|x\|_1 = \sum_{i=1}^n |x_i|$ (norme 1)
- $\|x\|_2 = \left(\sum_{i=1}^n |x_i|^2 \right)^{\frac{1}{2}}$ (norme 2, ou norme euclidienne)
- $\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$ (norme infinie)

Plus généralement : $\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}}$ (norme p).

La démonstration du fait qu’il s’agit d’une norme utilise les inégalités suivantes :

- Inégalité de Hölder : si p et q sont deux nombres vérifiant $p > 1$ et l’égalité $\frac{1}{p} + \frac{1}{q} = 1$ (ce qui entraîne $q > 1$), alors

$$\sum_{i=1}^n |x_i y_i| \leq \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}} \left(\sum_{i=1}^n |y_i|^q \right)^{\frac{1}{q}}.$$

Pour $p = q = 2$, cela redonne l’inégalité de Cauchy-Schwarz.

- Inégalité de Minkowski :

$$\left(\sum_{i=1}^n |x_i + y_i|^p \right)^{\frac{1}{p}} \leq \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}} + \left(\sum_{i=1}^n |y_i|^p \right)^{\frac{1}{p}}.$$

1.2. NORMES

13

Dans \mathbb{R}^n et \mathbb{C}^n , toutes les les normes sont *équivalentes* (deux normes $\| \cdot \|$ et $\| \cdot \|'$ sont équivalentes sur un espace vectoriel E s'il existe deux constantes strictement positives C et C' telles que, pour tout x dans E : $C\|x\| \leq \|x\|' \leq C'\|x\|$).

On peut également s'intéresser aux normes matricielles. On appelle \mathcal{A}_n l'anneau des matrices carrées d'ordre n à coefficients dans \mathbb{R} ou \mathbb{C} . On appelle *norme matricielle* une application de \mathcal{A}_n dans \mathbb{R}^+ notée $\| \cdot \|$ qui vérifie les propriétés suivantes :

- pour toute matrice A de \mathcal{A}_n , $\|A\| = 0 \Leftrightarrow A = 0$
- pour tout α de \mathbb{R} (ou \mathbb{C}) et pour tout A de \mathcal{A}_n , $\|\alpha A\| = |\alpha| \|A\|$
- pour toutes matrices A et B de \mathcal{A}_n , $\|A + B\| \leq \|A\| + \|B\|$
- pour toutes matrices A et B de \mathcal{A}_n , $\|A \times B\| \leq \|A\| \times \|B\|$.

On peut très facilement construire des normes matricielles à partir de normes vectorielles : elles sont dites alors *normes matricielles subordonnées*. Pour cela, on peut définir $\|A\|$ par les formules équivalentes suivantes:

$$\|A\| = \sup_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \sup_{\|x\|=1} \|Ax\| = \sup_{0 < \|x\| \leq 1} \frac{\|Ax\|}{\|x\|}.$$

L'inégalité suivante découle de ces définitions :

$$\text{pour tout vecteur } x, \|Ax\| \leq \|A\| \|x\|.$$

Les normes matricielles subordonnées aux normes les plus usuelles que nous avons décrites plus haut sont donc, pour $A = (a_{i,j})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}}$:

- $\|A\|_1 = \sup_{x \neq 0} \frac{\|Ax\|_1}{\|x\|_1} = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$
- $\|A\|_2 = \sup_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \sqrt{\rho(A^*A)} = \|A^*\|_2$ où $\rho(A^*A)$ représente le plus grand module des valeurs propres de A^*A (rayon spectral de A^*A)
- $\|A\|_\infty = \sup_{x \neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty} = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$.

La norme $\| \cdot \|_2$ est invariante par transformation unitaire : si U est une matrice unitaire, c'est-à-dire vérifie $U^*U = I$, on a alors

$$\|A\|_2 = \|AU\|_2 = \|UA\|_2 = \|U^*AU\|_2.$$

Si A est normale, c'est-à-dire si A vérifie $A^*A = AA^*$ (en particulier si A est hermitienne ou symétrique), alors $\|A\|_2 = \rho(A)$.

Si A est unitaire ou orthogonale, $\|A\|_2 = 1$.

Remarque : $\|A\|_1$ et $\|A\|_\infty$ sont faciles à calculer mais pas $\|A\|_2$.

Théorème 1.2.

- Soit $\|\cdot\|$ une norme subordonnée ; soit B vérifiant $\|B\| < 1$. Alors $I + B$ est inversible et $\|(I + B)^{-1}\| \leq \frac{1}{1 - \|B\|}$.
- Si une matrice de la forme $I + B$ n'est pas inversible, alors, pour toute norme, subordonnée ou non, $\|B\| \geq 1$.

Exemple de norme non subordonnée : la norme euclidienne

Cette norme est définie par : $\|A\|_E = \left(\sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2 \right)^{\frac{1}{2}} = \sqrt{\text{trace}(A^*A)}$ (on rappelle que la trace d'une matrice est la somme de ses termes diagonaux). La norme $\|A\|_E$ est invariante par transformation unitaire ; autrement dit, si $U^*U = I$, alors $\|A\|_E = \|AU\|_E = \|UA\|_E = \|U^*AU\|_E$.
De plus : $\|A\|_2 \leq \|A\|_E \leq \sqrt{n}\|A\|_2$.

Théorème 1.3. Soit $\|\cdot\|$ une norme quelconque (subordonnée ou non) ; on a : $\rho(A) \leq \|A\|$ et, pour tout $\varepsilon > 0$, il existe une norme subordonnée $\|\cdot\|_{A,\varepsilon}$ vérifiant $\|A\|_{A,\varepsilon} \leq \rho(A) + \varepsilon$.

Chapitre 2

Problèmes de l’analyse numérique

Les deux problèmes principaux que nous allons étudier dans la suite de ce cours sont la résolution des systèmes linéaires et le calcul des valeurs propres et vecteurs propres des matrices. Lorsqu’on applique les méthodes de l’analyse numérique à des problèmes de calcul, il faut prendre en compte deux types de « qualité ». Il s’agit d’une part de l’aspect que l’on appelle *complexité*, c’est-à-dire du nombre d’opérations élémentaires à effectuer pour obtenir un résultat, mais aussi il faut savoir déterminer si la solution est acceptable ou non ; en effet, on peut commettre deux sortes d’erreurs : d’une part, les erreurs d’arrondi, dues à la précision des calculs et, d’autre part, les erreurs dites de troncature, lorsque l’on utilise des méthodes itératives, alors que l’on s’arrête bien sûr après un nombre fini d’itérations.

2.1 Erreurs

Erreur d’arrondi : erreur due au codage où le nombre de chiffres représentant un réel est limité. Si le nombre est codé sur t bits pour la mantisse, l’erreur sur la mantisse est majorée par 2^{-t} .

Erreur de troncature : dans les méthodes itératives, le calcul de la limite nécessiterait *a priori* un nombre infini d’itérations. Comme on arrête forcément les calculs après un nombre k_0 d’itérations, on commet une erreur de troncature mesurée par $\|x^\infty - x^{k_0}\|$, où x^∞ représente la limite, x^{k_0} le résultat obtenu à la k_0^e itération et $\|\cdot\|$ une norme donnée, quand on arrête la méthode itérative (en fait, x^∞ est inconnu, ce qui ne permet pas d’estimer

l'erreur).

2.2 Conditionnement d'un système linéaire

Dans tout ce qui suit, on considère un système linéaire écrit sous la forme matricielle $Ax = b$. Avant de rentrer dans le détail des méthodes, qui feront l'objet du chapitre suivant, nous allons traiter d'un paramètre important des systèmes linéaires : il s'agit de leur *conditionnement*, lequel est attaché à la matrice A du système. Le plus souvent, dans la pratique, les coefficients de A , comme les composantes du vecteur b , sont les résultats de mesure et sont donc entachés d'une certaine erreur. Il est essentiel de voir comment une petite modification de A ou de b influe, indépendamment de la méthode utilisée, sur la solution supposée exacte du système.

Considérons le système suivant :

$$\begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix} \text{ de solution } \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

Considérons maintenant le système perturbé en modifiant légèrement le vecteur du second membre, la matrice restant inchangée :

$$\begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix} \begin{pmatrix} x_1 + \delta x_1 \\ x_2 + \delta x_2 \\ x_3 + \delta x_3 \\ x_4 + \delta x_4 \end{pmatrix} = \begin{pmatrix} 32,1 \\ 22,9 \\ 33,1 \\ 30,9 \end{pmatrix} \text{ de solution } \begin{pmatrix} 9,2 \\ -12,6 \\ 4,5 \\ -1,1 \end{pmatrix}.$$

On constate qu'une erreur relative de l'ordre de $1/300$ sur le second membre entraîne une erreur relative de l'ordre de 10 sur plusieurs coordonnées de la solution du système, et donc une amplification des erreurs relatives de l'ordre de 3000.

Considérons maintenant de légères modifications sur la matrice avec le système :

$$\begin{pmatrix} 10 & 7 & 8,1 & 7,2 \\ 7,08 & 5,04 & 6 & 5 \\ 8 & 5,98 & 9,89 & 9 \\ 6,99 & 4,99 & 9 & 9,98 \end{pmatrix} \begin{pmatrix} x_1 + \delta x_1 \\ x_2 + \delta x_2 \\ x_3 + \delta x_3 \\ x_4 + \delta x_4 \end{pmatrix} = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix} \text{ de solution } \begin{pmatrix} -81 \\ 137 \\ -34 \\ 22 \end{pmatrix}.$$

2.2. CONDITIONNEMENT D'UN SYSTÈME LINÉAIRE

17

On constate ici aussi que de petites variations des éléments de la matrice modifient considérablement la solution du système linéaire.

Supposons, toutes choses étant égales par ailleurs, que l'on considère le système : $A(x + \delta x) = b + \delta b$, et supposons la matrice A inversible. On voit que l'on a $\delta x = A^{-1}\delta b$. Si on choisit alors une norme matricielle $\| \cdot \|$, subordonnée à une norme vectorielle, on trouve $\|\delta x\| \leq \|A^{-1}\| \|\delta b\|$ et, de plus, $\|b\| \leq \|A\| \|x\|$ de sorte que l'on a sur x une erreur relative $\frac{\|\delta x\|}{\|x\|}$ majorée par $\|A\| \|A^{-1}\| \frac{\|\delta b\|}{\|b\|}$. On appelle *conditionnement* de la matrice A (relativement à la norme $\| \cdot \|$) la quantité $\|A\| \|A^{-1}\|$, ce que l'on note $\text{cond}_{\| \cdot \|}(A)$ ou, plus simplement, $\text{cond}(A)$.

On pourrait prouver de même que si l'on apporte maintenant une petite variation aux coefficients de A , de sorte que cette matrice devienne $A + \delta A$, alors $\frac{\|\delta x\|}{\|x + \delta x\|}$ est majorée par $\|A\| \|A^{-1}\| \frac{\|\delta A\|}{\|A\|}$.

Ces deux majorations prouvent l'intérêt du conditionnement. Une matrice est d'autant mieux conditionnée que son conditionnement (qui est toujours supérieur ou égal à 1) est proche de 1.

Le théorème suivant donne d'autres renseignements sur le conditionnement d'une matrice.

Théorème 2.1. Soit A une matrice inversible. On a alors :

1. $\text{cond}(A) \geq 1$
2. $\text{cond}(A) = \text{cond}(A^{-1})$
3. pour tout $\alpha \neq 0$, $\text{cond}(\alpha A) = \text{cond}(A)$
4. en notant cond_2 le conditionnement associé à $\| \cdot \|_2$ et en notant respectivement $\mu_1(A)$ et $\mu_n(A)$ la plus petite et la plus grande des valeurs singulières de A , $\text{cond}_2(A) = \frac{\mu_n(A)}{\mu_1(A)}$
5. si A est normale (c'est-à-dire vérifie $AA^* = A^*A$), $\text{cond}_2(A) = \frac{\max_i |\lambda_i(A)|}{\min_i |\lambda_i(A)|}$
où les $\lambda_i(A)$ représentent les valeurs propres de A
6. si A est unitaire ou orthogonale, $\text{cond}_2(A) = 1$

7. $\text{cond}_2(A)$ est invariant par transformation unitaire ou orthogonale :
 si $UU^* = I$, alors $\text{cond}_2(A) = \text{cond}_2(AU) = \text{cond}_2(UA) = \text{cond}_2(U^*AU)$,
 si $OO^t = I$, alors $\text{cond}_2(A) = \text{cond}_2(AO) = \text{cond}_2(OA) = \text{cond}_2(O^tAO)$.

Calculons par exemple le conditionnement de la matrice utilisée précédemment :

$$A = \begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix}.$$

Cette matrice a pour valeurs propres approchées :

$$\lambda_1 \approx 0,01015 < \lambda_2 \approx 0,8431 < \lambda_3 \approx 3,858 < \lambda_4 \approx 30,2887.$$

On a ainsi : $\text{cond}_2(A) = \frac{\lambda_4}{\lambda_1} \approx 2984$. La matrice A a donc un très mauvais conditionnement, ce qui explique la sensibilité aux erreurs des systèmes linéaires définis avec la matrice A .

Comme pour tout $\alpha \neq 0$, $\text{cond}(\alpha A) = \text{cond}(A)$, on ne peut espérer diminuer le conditionnement de A en multipliant tous ses éléments par un même nombre. En revanche, on peut le faire en multipliant par exemple chaque ligne (et/ou chaque colonne) par un coefficient approprié ; c'est là le problème de l'*équilibrage d'une matrice*, qui peut s'énoncer comme suit : étant donnée une matrice A , déterminer deux matrices diagonales inversibles D_1 et D_2 vérifiant : $\text{cond}(D_1AD_2) = \inf_{\Delta_1, \Delta_2 \text{ diagonales inversibles}} \text{cond}(\Delta_1A\Delta_2)$.

On résout alors $Ax = b$ en deux étapes :

- résolution de $D_1AD_2y = D_1b$
- résolution de $x = D_2y$.

En pratique, le conditionnement n'est pas une fonction simple des éléments de D_1 et D_2 ; on essaie plutôt de minimiser le rapport entre le plus grand et le plus petit élément non nul de $A' = \Delta_1A\Delta_2$. Posons $E = \{(i, j) \text{ avec } 1 \leq i \leq n, 1 \leq j \leq n \text{ et } a_{ij} \neq 0\}$. On cherche deux matrices Δ_1 et Δ_2 diagonales et inversibles qui minimisent le rapport :

$$\frac{\max_{(i,j) \in E} |a'_{ij}|}{\min_{(i,j) \in E} |a'_{ij}|}.$$

2.2. CONDITIONNEMENT D'UN SYSTÈME LINÉAIRE

19

On considère maintenant le cas où A est une matrice réelle. En notant x_i le i^e élément de la diagonale de Δ_1 et y_i le i^e élément de la diagonale de Δ_2 , on a : $a'_{ij} = x_i a_{ij} y_j$. On passe aux logarithmes en posant $\alpha_{ij} = \ln |a_{ij}|$, $u_i = \ln |x_i|$, $v_j = \ln |y_j|$. Le problème devient :

$$\text{minimiser } u_i, v_j \text{ avec } (i, j) \in E \left[\max_{(i, j) \in E} (\alpha_{ij} + u_i + v_j) - \min_{(i, j) \in E} (\alpha_{ij} + u_i + v_j) \right],$$

ce qui se réécrit comme le programme linéaire suivant (car on peut, par une translation des valeurs, se restreindre aux solutions où le minimum sur les u_i et v_j de $\alpha_{ij} + u_i + v_j$ vaut 0) :

$$\begin{cases} \text{minimiser } z \\ \text{avec, pour tout } (i, j) \in E, \quad 0 \leq \alpha_{ij} + u_i + v_j \leq z \\ u_i \text{ et } v_j \text{ de signes quelconques.} \end{cases}$$

Chapitre 3

Résolution de systèmes linéaires

3.1 Généralités

Le problème auquel on s'intéresse peut se formuler de la façon suivante.

Problème : Soient $A = (a_{i,j})$ une matrice carrée inversible de dimension n , $x = (x_i)$ et $b = (b_i)$ deux vecteurs colonnes de dimension n ; résoudre par rapport à x le système $Ax = b$.

Remarques :

1. Les méthodes numériques de résolution n'utilisent généralement pas le calcul de A^{-1} .
2. Si A est sous forme triangulaire supérieure (les éléments sous la diagonale principale sont tous nuls) avec des termes diagonaux non nuls, alors la résolution est aisée. On commence par résoudre la dernière relation, qui est une équation linéaire en la seule variable x_n , on reporte cette valeur dans l'avant-dernière relation qui devient une équation en x_{n-1} , et on continue ainsi de proche en proche jusqu'à x_1 . Cette méthode, dite *méthode de remontée* et résumée ci-dessous, nécessite $n(n-1)/2$ additions, $n(n-1)/2$ multiplications et n divisions.

$$\left\{ \begin{array}{lclclcl} a_{1,1}x_1 + & \dots & + & a_{1,n-1}x_{n-1} & + & a_{1,n}x_n & = & b_1 \\ & & & \dots & & & & \\ & & & a_{n-1,n-1}x_{n-1} & + & a_{n-1,n}x_n & = & b_{n-1} \\ & & & & & a_{n,n}x_n & = & b_n \end{array} \right.$$

$$\Rightarrow \begin{cases} x_n = \frac{b_n}{a_{n,n}} \\ x_{n-1} = \frac{b_{n-1} - a_{n-1,n}x_n}{a_{n-1,n-1}} \\ \dots \\ x_1 = \frac{b_1 - a_{1,2}x_2 - \dots - a_{1,n}x_n}{a_{1,1}} \end{cases}$$

3.2 Méthode de Gauss

La méthode de Gauss est utilisée lorsque la matrice A est quelconque. Le principe en est le suivant :

- À l’aide de combinaisons linéaires entre les lignes de A , on élimine successivement certaines inconnues des relations, pour obtenir une forme $(MA)x = Mb$ où MA est une matrice triangulaire supérieure. Remarquons qu’en fait on ne calcule pas M , mais qu’on construit directement MA et Mb .
- On résout $(MA).x = M.b$ par une méthode de remontée.

3.2.1 Étape d’élimination

- On choisit dans la première colonne un coefficient $a_{i,1}$ différent de 0 ; il en existe toujours puisque la matrice est inversible. Cet élément constitue le *pivot*.
- Si le pivot n’est pas en première ligne, on échange la ligne du pivot avec la première ligne.
- Par des combinaisons linéaires bien choisies, obtenues en retranchant à chaque ligne la première ligne multipliée par le bon coefficient, on annule tous les termes de la colonne du pivot situés sous la diagonale.
- On obtient alors une matrice A' dont la première colonne n’a que des 0 sous le premier terme qui, lui, est non nul.
- On considère la matrice obtenue en supprimant la première ligne et la première colonne de A' . On réitère le procédé sur cette nouvelle matrice.
- On arrête ce procédé quand la matrice obtenue est de dimension 1.

3.2. MÉTHODE DE GAUSS

23

— En remplaçant les lignes et colonnes supprimées au fur et à mesure, on obtient une matrice triangulaire.

Remarque : Le déterminant de A s’obtient par le produit des pivots multiplié par $(-1)^p$, où p représente le nombre de fois que le pivot n’était pas sur la diagonale. Ceci sera généralisé plus loin, dans la partie concernant le choix du pivot.

Exemple 1

On considère le système :

$$\begin{cases} 2x_1 + x_2 - 3x_3 = 5 \\ 4x_1 + x_2 + 5x_3 = -1 \\ 10x_1 - 7x_2 + 13x_3 = -3 \end{cases}$$

Après la première itération, en ayant choisi comme pivot la valeur 2, en gras ci-dessus, on obtient :

$$\begin{cases} 2x_1 + x_2 - 3x_3 = 5 \\ -1x_2 + 11x_3 = -11 \\ -12x_2 + 28x_3 = -28 \end{cases}$$

Après la seconde itération (le pivot est le coefficient de la deuxième ligne, deuxième colonne et vaut -1), on obtient :

$$\begin{cases} 2x_1 + x_2 - 3x_3 = 5 \\ -x_2 + 11x_3 = -11 \\ -104x_3 = 104 \end{cases}$$

On applique alors une méthode de remontée, et l’on obtient successivement:

$$x_3 = -1, x_2 = \frac{-11 - 11x_3}{-1} = 0, x_1 = \frac{5 - x_2 + 3x_3}{2} = 1.$$

Remarque : comme les lignes n’ont pas été échangées, le déterminant de A est égal au déterminant de la matrice correspondant au dernier système. On a donc : $\det(A) = 2 \times (-1) \times (-104) = 208$.

Exemple 2

On considère le système :

$$\begin{cases} \mathbf{2}x_1 + x_2 - 3x_3 = -3 \\ 4x_1 + 2x_2 - x_3 = 4 \\ 6x_1 + 5x_2 + 8x_3 = 27 \end{cases}$$

Après la première itération, en ayant choisi comme pivot la valeur 2, en gras ci-dessus, on obtient :

$$\begin{cases} 2x_1 + x_2 - 3x_3 = -3 \\ 0x_2 + 5x_3 = 10 \\ \mathbf{2}x_2 + 17x_3 = 36 \end{cases}$$

Le pivot est maintenant nécessairement le coefficient de x_2 dans la dernière ligne (de valeur 2) ; on échange la deuxième et la troisième ligne ; on obtient :

$$\begin{cases} 2x_1 + x_2 - 3x_3 = -3 \\ \mathbf{2}x_2 + 17x_3 = 36 \\ 0x_2 + 5x_3 = 10 \end{cases}$$

Le coefficient de x_2 dans la dernière ligne étant nul, il ne reste plus qu'à effectuer la remontée :

$$x_3 = 10/5 = 2, x_2 = \frac{36 - 17x_3}{2} = 1, x_1 = \frac{-3 - x_2 + 3x_3}{2} = 1.$$

Remarque : les lignes ayant été échangées une fois, le déterminant de A est égal au déterminant de la matrice correspondant au dernier système multiplié par -1 . On a donc : $\det(A) = (-1) \times 2 \times 2 \times 5 = -20$.

3.2.2 Choix du pivot

À cause des erreurs d'arrondi, le choix du pivot est important ; en effet, un pivot trop petit en valeur absolue peut conduire à de mauvaises solutions du fait de la division par le pivot. Trois stratégies sont en fait possibles.

- *Stratégie par défaut* : on choisit comme pivot le terme situé à l'intersection de la colonne courante et de la ligne courante, ce qui n'est possible que si ce terme n'est pas nul (sinon, il faut procéder à des échanges de lignes ou de colonnes).
- *Pivot partiel* : on choisit dans la colonne courante le terme de plus grande valeur absolue situé sous la diagonale ou sur celle-ci. C'est souvent ce choix qui est retenu quand on programme la méthode.

3.2. MÉTHODE DE GAUSS

25

- *Pivot total* : on choisit le terme de plus grande valeur absolue de la matrice résiduelle, c’est-à-dire, si on est à l’étape $n - k + 1$, la matrice constituée des k dernières lignes et des k dernières colonnes. Cette méthode, plus coûteuse en temps, est en fait peu utilisée.

Remarque : le déterminant de A s’obtient par le produit des pivots multiplié par $(-1)^p$, où p représente le nombre de fois que l’on a effectué des échanges de lignes ou de colonnes.

3.2.3 Complexité

On peut évaluer le nombre d’opérations nécessaires pour la méthode de Gauss ; dans le cas où on ne choisit pas le pivot, on effectue en tout environ $\frac{n^3}{3}$ additions, autant de multiplications, $\frac{n^2}{2}$ divisions et donc au total un nombre d’opérations arithmétiques équivalent à $\frac{2n^3}{3}$.

3.2.4 Variante : la méthode de Gauss-Jordan

Par rapport à la méthode de Gauss, la seule différence apportée par la méthode de Gauss-Jordan est que, dans la phase d’élimination, on élimine également les termes situés au-dessus de la diagonale. On obtient ainsi une matrice diagonale. Cette méthode est notamment utilisée pour le calcul de l’inverse d’une matrice. On résout alors simultanément les n systèmes linéaires $Ax_j = e_j$, l’inconnue étant le vecteur colonne x_j (le j^{e} vecteur de la matrice inverse), les e_j constituant les vecteurs de la base canonique de \mathbb{R}^n .

Exemple : Calcul de l’inverse de $A = \begin{pmatrix} 1 & -3 & 14 \\ 1 & -2 & 10 \\ -2 & 4 & -19 \end{pmatrix}$.

On résout les trois systèmes :

$$\left\{ \begin{array}{rrcrcl} x_1 & - & 3x_2 & + & 14x_3 & = & 1 & | & 0 & | & 0 \\ x_1 & - & 2x_2 & + & 10x_3 & = & 0 & | & 1 & | & 0 \\ -2x_1 & + & 4x_2 & - & 19x_3 & = & 0 & | & 0 & | & 1 \end{array} \right.$$

Première itération (ici, avec pivot partiel) : on échange la première et la troisième lignes, ce qui donne, avec le pivot en haut à gauche (en gras) :

$$\left\{ \begin{array}{rrcrcl} -\mathbf{2}x_1 & + & 4x_2 & - & 19x_3 & = & 0 & | & 0 & | & 1 \\ x_1 & - & 2x_2 & + & 10x_3 & = & 0 & | & 1 & | & 0 \\ x_1 & - & 3x_2 & + & 14x_3 & = & 1 & | & 0 & | & 0 \end{array} \right.$$

On élimine les termes de la première colonne sauf le terme diagonal. On obtient :

$$\left\{ \begin{array}{rrcr} -2x_1 & + & 4x_2 & - & 19x_3 & = & 0 & \left| \begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right| & \begin{array}{c} 1 \\ 1/2 \\ 1/2 \end{array} \\ & & 0x_2 & + & 1/2x_3 & = & 0 & & \\ & & - & x_2 & + & 9/2x_3 & = & 1 & \end{array} \right.$$

Deuxième itération (maintenant, avec pivot total pour illustrer cette variante) : le plus grand coefficient en valeur absolue étant $9/2$, en bas à droite, on échange la deuxième et la troisième lignes ainsi que la deuxième et la troisième colonnes. On obtient :

$$\left\{ \begin{array}{rrcr} -2x_1 & - & 19x_3 & + & 4x_2 & = & 0 & \left| \begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right| & \begin{array}{c} 1 \\ 1/2 \\ 1/2 \end{array} \\ & & 9/2x_3 & - & x_2 & = & 1 & & \\ & & 1/2x_3 & + & 0x_2 & = & 0 & & \end{array} \right.$$

On élimine les termes de la deuxième colonne sauf le terme diagonal :

$$\left\{ \begin{array}{rrcr} -2x_1 & & & - & 2/9x_2 & = & 38/9 & \left| \begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right| & \begin{array}{c} 28/9 \\ 1/2 \\ 4/9 \end{array} \\ & & 9/2x_3 & - & x_2 & = & 1 & & \\ & & & & 1/9x_2 & = & -1/9 & & \end{array} \right.$$

Troisième et dernière itération ; le pivot ne peut être que l'élément de la ligne non encore traitée : il s'agit du $1/9$ en bas à droite. On obtient :

$$\left\{ \begin{array}{rrcr} -2x_1 & & & = & 4 & \left| \begin{array}{c} 2 \\ 9 \\ 1 \end{array} \right| & \begin{array}{c} 4 \\ 9/2 \\ 4/9 \end{array} \\ & & 9/2x_3 & = & 0 & & \\ & & & & 1/9x_2 & = & -1/9 \end{array} \right.$$

On peut maintenant résoudre les trois systèmes immédiatement :

$$\left\{ \begin{array}{rrcr} x_1 & = & -2 & \left| \begin{array}{c} -1 \\ 9 \\ 2 \end{array} \right| & \begin{array}{c} -2 \\ 4 \\ 1 \end{array} \\ x_2 & = & -1 & & \\ x_3 & = & 0 & & \end{array} \right.$$

On déduit l'inverse de A de ces calculs : $A^{-1} = \begin{pmatrix} -2 & -1 & -2 \\ -1 & 9 & 4 \\ 0 & 2 & 1 \end{pmatrix}$.

Comme il y a deux échanges de lignes et un échange de colonnes, le déterminant de A vaut : $(-1)^3 \times (-2) \times \frac{9}{2} \times \frac{1}{9} = 1$.

3.3 Factorisation LU

Dans la méthode de Gauss, on transforme le système $Ax = b$ en un système $MAx = Mb$ où MA est une matrice triangulaire supérieure, que nous noterons U (pour *upper*). Supposons que, dans cette construction, le pivot se trouve toujours sur la diagonale (stratégie par défaut sans échange de ligne ou de colonne), ce qui implique que le terme qui apparaît dans la case d’indices (k, k) après $k - 1$ ($1 \leq k \leq n$) étapes ne vaut jamais 0 (c’est largement le cas général).

Soit $1 \leq k \leq n$. Notons A_k la matrice du système obtenue après $k - 1$ étapes, c’est-à-dire au début de la k^e étape. Cette matrice a des 0 sous la diagonale pour les $k - 1$ premières colonnes (i.e. pour tout couple d’indices (s, t) avec $1 \leq t \leq k - 1, s > t$) et, par hypothèse, $(A_k)_{k,k}$ est non nul. On a $A_1 = A$ et $A_n = MA = U$. Pour $1 \leq i \leq n$, posons $\alpha_i = (A_k)_{i,k}$; ainsi, la k^e colonne de A_k (la colonne du pivot) est :

$$(\alpha_1, \dots, \alpha_k \neq 0, \dots, \alpha_n)^t.$$

Pour $1 \leq k \leq n - 1$, soit E_k la matrice qui possède des 1 sur la diagonale et ailleurs des 0 sauf pour $(E_k)_{i,k}$ avec $i > k$ (partie de la k^e colonne située sous la diagonale) : pour $i > k$, on pose $(E_k)_{i,k} = -\frac{\alpha_i}{\alpha_k}$. Cette matrice E_k est donc triangulaire inférieure et ne diffère de la matrice identité que par sa k^e colonne :

$$E_k = \begin{pmatrix} 1 & 0 & \dots & & & \\ 0 & 1 & 0 & \dots & & \\ & & \dots & 0 & \dots & \\ & \dots & 0 & 1 & 0 & \dots \\ & & & -\frac{\alpha_{k+1}}{\alpha_k} & 1 & 0 & \dots \\ & & & \dots & & & \\ & & & -\frac{\alpha_n}{\alpha_k} & 0 & \dots & 1 \end{pmatrix}.$$

Par la méthode de Gauss, on passe alors de la matrice A_k à la matrice A_{k+1} en multipliant A_k à gauche par la matrice E_k : $A_{k+1} = E_k A_k$. D’où, par une récurrence immédiate, $A_n = E_{n-1} E_{n-2} \dots E_1 A_1$. Comme on a $A_n = MA$ et $A_1 = A$, on en déduit la relation $M = E_{n-1} E_{n-2} \dots E_1$. Le produit de matrices triangulaires inférieures dont la diagonale ne contient que des 1 étant aussi une matrice triangulaire inférieure avec une diagonale de 1, la matrice M est elle-même triangulaire inférieure avec une diagonale de 1. Elle est donc inversible.

La relation $MA = U$ donne : $A = M^{-1}U$. On pose $L = M^{-1}$. D'où $A = LU$. L'inverse d'une matrice triangulaire inférieure avec une diagonale de 1 étant aussi une matrice triangulaire inférieure avec une diagonale de 1, on conclut que L est elle-même une matrice triangulaire inférieure avec une diagonale de 1 (d'où la notation L , pour *lower*).

On pose maintenant, pour $1 \leq k < i \leq n$, $\beta_{i,k} = \frac{\alpha_i}{\alpha_k}$: $\beta_{i,k}$ est le facteur par lequel, à la k^e étape, on multiplie la k^e ligne du système (la ligne du pivot) pour obtenir, en soustrayant cette k^e ligne ainsi multipliée à la i^e ligne, la nouvelle i^e ligne. On vérifie facilement l'expression de E_k^{-1} :

$$E_k^{-1} = \begin{pmatrix} 1 & 0 & \dots & & & \\ 0 & 1 & 0 & \dots & & \\ & & \dots & 0 & \dots & \\ & \dots & 0 & 1 & 0 & \dots \\ & & & \beta_{k+1,k} & 1 & 0 & \dots \\ & & & \dots & & & \\ & & & \beta_{n,k} & 0 & \dots & 1 \end{pmatrix}.$$

Or, on a $L = M^{-1} = E_1^{-1} \dots E_{n-2}^{-1} E_{n-1}^{-1}$. On vérifie facilement aussi le résultat suivant :

$$L = E_1^{-1} \dots E_{n-2}^{-1} E_{n-1}^{-1} = \begin{pmatrix} 1 & 0 & \dots & & & \\ \beta_{2,1} & 1 & 0 & \dots & & \\ \beta_{3,1} & \beta_{3,2} & 1 & 0 & \dots & \\ & & & \dots & & \\ \beta_{n,1} & \beta_{n,2} & \beta_{n,3} & \dots & \beta_{n,n-1} & 1 \end{pmatrix}.$$

On obtient ainsi la *décomposition* (ou *factorisation*) LU classique de A : $A = LU$, avec L matrice triangulaire inférieure dont la diagonale ne contient que des 1 et U matrice triangulaire supérieure.

Exemple

On reprend l'exemple 1 de la méthode de Gauss avec : $A = \begin{pmatrix} 2 & 1 & -3 \\ 4 & 1 & 5 \\ 10 & -7 & 13 \end{pmatrix}$.

La résolution du système donne l'expression suivante pour U :

$$U = \begin{pmatrix} 2 & 1 & -3 \\ 0 & -1 & 11 \\ 0 & 0 & -104 \end{pmatrix}.$$

3.3. FACTORISATION LU

29

Les paragraphes ci-dessus et l’observation des facteurs utilisés pour faire apparaître les 0 sous la diagonale donne l’expression suivante pour L :

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 5 & 12 & 1 \end{pmatrix}.$$

Les considérations précédentes reposent sur l’hypothèse selon laquelle, dans l’application de la méthode de Gauss, le pivot se trouve sur la diagonale (*cf.* plus haut). Le théorème suivant donne une condition suffisante pour que cette hypothèse soit vérifiée.

Théorème 3.1. (d’existence de la factorisation LU) Soit $A = (a_{ij})$ une matrice carrée (invertible) telle que, pour tout k compris entre 1 et n , la sous-matrice $\begin{pmatrix} a_{11} & \dots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \dots & a_{kk} \end{pmatrix}$ soit invertible. Alors, la factorisation $A = LU$ est possible (plus précisément, les pivots successifs peuvent toujours être pris sur la diagonale, sans échange de lignes). De plus, on peut choisir $(L)_{ii} = 1$ et la décomposition est alors unique : c’est la factorisation LU classique (celle qui est adoptée par défaut).

En fait, on peut montrer que si la factorisation LU échoue (c’est-à-dire si les pivots ne peuvent pas être toujours choisis sur la diagonale sans échange de lignes), on peut permuter au départ les lignes de la matrice A pour obtenir une matrice A' pour laquelle la factorisation LU est possible.

Lorsque l’on doit résoudre plusieurs systèmes linéaires de même matrice A , on calcule la factorisation LU lors de la résolution du premier de ces systèmes. La résolution de tout système ultérieur $Ax = b$ se ramène à la résolution de deux systèmes de matrices triangulaires : le système $Ly = b$ puis le système $Ux = y$ (on notera ainsi qu’il est inutile de connaître M explicitement, dont le calcul n’est pas nécessairement aisé). Chaque système ne prend plus alors que $n(n-1)$ additions, $n(n-1)$ multiplications et $2n$ divisions.

3.4 Méthode de Cholesky

La méthode de Cholesky donne une factorisation intéressante dans le cas des matrices symétriques définies positives. Dans ce cas, on peut choisir une factorisation LU avec $U = L^t$ en renonçant néanmoins à avoir des termes diagonaux tous égaux à 1 dans L .

Théorème 3.2. Soit A une matrice symétrique définie positive. Il existe une matrice triangulaire B vérifiant $A = BB^t$. De plus, on peut imposer que les éléments diagonaux de la matrice B soient tous strictement positifs et la factorisation $A = BB^t$ est alors unique : c’est la factorisation classique (celle qui est adoptée par défaut).

En pratique, on calcule la matrice $B = \begin{pmatrix} b_{11} & 0 & \dots & 0 \\ b_{21} & b_{22} & 0 & \dots & 0 \\ & & \dots & & \\ b_{n1} & b_{n2} & \dots & & b_{nn} \end{pmatrix}$ colonne

par colonne, à partir des égalités la définissant :

$$\text{pour } 1 \leq i \leq j \leq n, a_{ij} = \sum_{k=1}^i b_{ik}b_{jk} = a_{ji}.$$

— Pour la première colonne, la formule donne

$$\text{— } b_{11} = \sqrt{a_{11}}$$

$$\text{— pour } 2 \leq i \leq n, b_{i1} = \frac{a_{i1}}{b_{11}}.$$

— Pour $2 \leq j \leq n$,

$$\text{— sur la diagonale, } b_{jj} = \sqrt{a_{jj} - \sum_{k=1}^{j-1} b_{jk}^2}$$

$$\text{— pour } j < i \leq n, b_{ij} = \frac{a_{ij} - \sum_{k=1}^{j-1} b_{ik}b_{jk}}{b_{jj}}.$$

Remarques.

1. La preuve du théorème précédent permettrait de montrer que les b_{ij} ainsi obtenus sont bien définis, grâce au fait que A est définie positive.

2. Le déterminant de la matrice A peut se calculer facilement :

$$\det(A) = (b_{11}b_{22}\dots b_{nn})^2.$$

Un système $Ax = b$ devient alors $BB^tx = b$. Pour résoudre le système, on résout $By = b$ puis $B^tx = y$.

3.4. MÉTHODE DE CHOLESKY

31

Complexité. Au total (la factorisation et les deux résolutions), on a effectué de l'ordre de $n^3/6$ additions, $n^3/6$ multiplications, $n^2/2$ divisions, n extractions de racines carrées, soit de l'ordre de $n^3/3$ opérations, c'est-à-dire environ la moitié des opérations mises en œuvre par la méthode de Gauss. On a donc intérêt à appliquer la méthode de Cholesky plutôt que la méthode de Gauss quand A est symétrique définie positive.

Exemple.

Considérons le système suivant :

$$\begin{cases} 4x_1 - 2x_2 &= 4 \\ -2x_1 + 2x_2 + 3x_3 &= -8 \\ 3x_2 + 10x_3 &= -20 \end{cases}$$

La matrice A correspondante est : $A = \begin{pmatrix} 4 & -2 & 0 \\ -2 & 2 & 3 \\ 0 & 3 & 10 \end{pmatrix}$. Cette matrice est symétrique définie positive. En effet, soit x un vecteur de \mathbb{R}^n représenté comme un vecteur colonne. On a alors :

$$\begin{aligned} x^t A x &= 4x_1^2 - 4x_1x_2 + 2x_2^2 + 6x_2x_3 + 10x_3^2 \\ &= (2x_1 - x_2)^2 + (x_2 + 3x_3)^2 + x_3^2. \end{aligned}$$

Par conséquent, si x est non nul, $x^t A x$ est un réel strictement positif.

Première étape : on calcule B telle que $A = BB^t$ avec B triangulaire supérieure. L'application des formules précédentes donne :

$$b_{11} = \sqrt{a_{11}} = 2 ; b_{21} = \frac{a_{21}}{b_{11}} = -1 ; b_{31} = \frac{a_{31}}{b_{11}} = 0$$

$$b_{22} = \sqrt{a_{22} - \sum_{k=1}^1 b_{2k}^2} = \sqrt{2 - 1} = 1$$

$$b_{32} = \frac{a_{32} - \sum_{k=1}^1 b_{3k}b_{2k}}{b_{22}} = \frac{3 - 0 \times (-1)}{1} = 3$$

$$b_{33} = \sqrt{a_{33} - \sum_{k=1}^2 b_{3k}^2} = \sqrt{10 - 0 - 9} = 1.$$

$$\text{D'où : } B \begin{pmatrix} 2 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} \text{ et } \det(A) = (2 \times 1 \times 1)^2 = 4.$$

Seconde étape : par la méthode de remontée, on résout les deux systèmes $By = b$ et $B^t x = y$.

Le système $By = b$ s'écrit :

$$\begin{cases} 2y_1 & & & = & 4 \\ -y_1 & + & y_2 & & = & -8 \\ & & 3y_2 & + & y_3 & = & -20 \end{cases}$$

qui a pour solution $y_1 = 2, y_2 = -6, y_3 = -2$.

Le système $B^t x = y$ s'écrit :

$$\begin{cases} 2x_1 & - & x_2 & & = & 2 \\ & & x_2 & + & 3x_3 & = & -6 \\ & & & & x_3 & = & -2 \end{cases}$$

qui a pour solution $x_3 = -2, x_2 = 0, x_1 = 1$.

La solution du système est donc : $x = \begin{pmatrix} 1 \\ 0 \\ -2 \end{pmatrix}$.

Si on avait un autre système à résoudre avec la même matrice A , seule la seconde étape serait appliquée.

Chapitre 4

Valeurs et vecteurs propres

Remarquons d’abord que la recherche des valeurs propres d’une matrice, au contraire du calcul de son inverse, est un problème difficile. Étant donné le polynôme $P(\lambda) = \lambda^n + a_1\lambda^{n-1} + \dots + a_{n-1}\lambda + a_n$, définissons la matrice :

$$\begin{pmatrix} -a_1 & -a_2 & -a_3 & \dots & -a_{n-1} & -a_n \\ 1 & 0 & & & & \\ 0 & 1 & 0 & & & \\ & 0 & 1 & 0 & & \\ & & & \dots & & \\ & & & 0 & 1 & 0 \\ & & & & 0 & 1 & 0 \end{pmatrix}$$

Cette matrice est dite « compagne du polynôme » P . Son polynôme caractéristique vaut $(-1)^n P(\lambda)$; la matrice a donc pour valeurs propres les racines de P . Or, d’après le théorème d’Abel, il est impossible de calculer les racines de tout polynôme à partir du degré 5 à l’aide d’un nombre fini d’applications des quatre opérations arithmétiques usuelles plus l’extraction de racines. Si une méthode de recherche de valeurs propres convergerait toujours en un nombre fini de ces opérations, il en serait alors de même de la recherche des racines d’une équation polynomiale quelconque, ce qui est contraire au résultat d’Abel. En revanche, il est courant, pour déterminer les racines d’un polynôme P , de chercher les valeurs propres de la compagne de P .

Pour calculer une approximation des valeurs propres d’une matrice A , l’idée de base est de rechercher une matrice semblable à A , c’est-à-dire de la forme $P^{-1}AP$, triangulaire ou diagonale, et dont la diagonale sera donc constituée des valeurs propres de A . Nous étudierons dans ce chapitre une

seule méthode, la méthode de Jacobi, qui s’applique au cas des matrices symétriques réelles. Rappelons que les valeurs propres d’une telle matrice sont réelles.

4.1 Méthode de Jacobi

Soit A une matrice symétrique réelle, soient deux indices p et q vérifiant $p < q$ tels que l’élément (non diagonal) a_{pq} soit non nul (s’il n’en existe pas, A est diagonale et les valeurs propres de A sont précisément les valeurs de la diagonale).

Soit θ un nombre réel ; on définit une matrice Ω dépendant de θ . La matrice Ω diffère de la matrice identité d’ordre n uniquement par les quatre coefficient suivants :

$$\Omega_{pp} = \Omega_{qq} = \cos \theta, \Omega_{pq} = \sin \theta, \Omega_{qp} = -\sin \theta.$$

La matrice Ω est représentée ci-dessous.

$$\Omega = \begin{pmatrix} 1 & 0 & & \dots & & 0 & 0 \\ 0 & 1 & & \dots & & 0 & 0 \\ & & \dots & \cos \theta & \dots & \sin \theta & \\ & & & & 1 & \dots & \\ & & & & \dots & \dots & \\ & & & & \dots & 1 & \dots \\ & & & -\sin \theta & \dots & \cos \theta & \\ & & \dots & & \dots & & \dots \\ 0 & 0 & & \dots & & 1 & 0 \\ 0 & 0 & & \dots & & 0 & 1 \end{pmatrix}.$$

La matrice Ω est orthogonale. C’est la matrice de rotation d’angle $-\theta$ dans le plan défini par les p^e et q^e vecteurs de base.

On pose : $B = \Omega^t A \Omega$. La matrice B , elle aussi symétrique, est semblable à la matrice A et admet donc les mêmes valeurs propres que A . On établit

4.1. MÉTHODE DE JACOBI

35

facilement les égalités suivantes :

$$\left\{ \begin{array}{l} \text{si } i \notin \{p, q\} \text{ et } j \notin \{p, q\}, \quad b_{ij} = b_{ji} = a_{ij} \\ \text{si } i \notin \{p, q\}, \quad b_{pi} = b_{ip} = a_{pi} \cos \theta - a_{qi} \sin \theta \\ \text{si } i \notin \{p, q\}, \quad b_{qi} = b_{iq} = a_{pi} \sin \theta + a_{qi} \cos \theta \\ b_{pp} = a_{pp} \cos^2 \theta + a_{qq} \sin^2 \theta - a_{pq} \sin 2\theta \\ b_{qq} = a_{pp} \sin^2 \theta + a_{qq} \cos^2 \theta + a_{pq} \sin 2\theta \\ b_{pq} = b_{qp} = a_{pq} \cos 2\theta + \frac{a_{pp} - a_{qq}}{2} \sin 2\theta. \end{array} \right.$$

On remarque l'équivalence $b_{pq} = 0 \Leftrightarrow \cot 2\theta = \frac{a_{qq} - a_{pp}}{2a_{pq}}$ (où \cot désigne la fonction trigonométrique cotangente). On essaie de faire en sorte d'avoir $b_{pq} = 0$ et on choisit donc θ pour qu'il vérifie la formule ci-dessus. Il y a quatre solutions dans l'intervalle $]-\pi, \pi]$, deux solutions successives différant de $\pi/2$. Il y a donc une unique solution dans l'intervalle $]-\frac{\pi}{4}, \frac{\pi}{4}]$, c'est la solution retenue.

Posons maintenant : $x = \frac{a_{qq} - a_{pp}}{2a_{pq}}, t = \tan \theta, s = \sin \theta, c = \cos \theta$. On rappelle les relations trigonométriques suivantes :

$$\cot 2\theta = \frac{\cos 2\theta}{\sin 2\theta} = \frac{\cos^2 \theta - \sin^2 \theta}{2 \sin \theta \cos \theta} = \frac{1 - t^2}{2t}.$$

On cherche à avoir : $x = \frac{1 - t^2}{2t}$; il en résulte que t doit vérifier l'équation : $t^2 + 2xt - 1 = 0$. Comme le produit des racines vaut -1 et que θ est dans l'intervalle $]-\frac{\pi}{4}, \frac{\pi}{4}]$, t est la racine de l'équation de plus petit module si les racines ne sont pas 1 et -1 , et vaut 1 si $x = 0$.

Comme on a $c > 0$, il vient $c = \frac{1}{\sqrt{1 + t^2}}$ et $s = ct = \frac{t}{\sqrt{1 + t^2}}$.

Les coefficients de la matrice B peuvent finalement être calculés par les formules suivantes, dans lesquelles t , c et s sont définis comme ci-dessus :

$$\begin{cases} \text{si } i \notin \{p, q\} \text{ et } j \notin \{p, q\}, b_{ij} = b_{ji} = a_{ij} \\ \text{si } i \notin \{p, q\}, b_{pi} = b_{ip} = ca_{pi} - sa_{qi} \\ \text{si } i \notin \{p, q\}, b_{qi} = b_{iq} = sa_{pi} + ca_{qi} \\ b_{pp} = a_{pp} - ta_{pq} \\ b_{qq} = a_{qq} + ta_{pq}. \end{cases}$$

Cette transformation, qui a le mérite d’annuler des éléments non diagonaux, peut en même temps rendre non nuls des éléments qui étaient précédemment nuls, comme le montre l’exemple 3 plus bas. Il y a cependant de bonnes raisons d’espérer, en réitérant le procédé, une convergence des matrices B obtenues vers une matrice diagonale, comme nous allons l’expliquer ci-dessous.

Théorème 4.1. . Soit A une matrice symétrique réelle et soit B la matrice obtenue à l’aide du procédé précédent. On a alors les relations :

$$\sum_{i=1}^n \sum_{j=1}^n a_{ij}^2 = \sum_{i=1}^n \sum_{j=1}^n b_{ij}^2,$$

$$\sum_{i=1}^n a_{ii}^2 + 2a_{pq}^2 = \sum_{i=1}^n b_{ii}^2.$$

Preuve. La première relation résulte de la conservation de la norme $\| \cdot \|_E$ par une transformation unitaire. Quant à la seconde, seuls les éléments des lignes et colonnes p et q sont modifiés. Les éléments diagonaux autres que a_{pp} et a_{qq} sont donc invariants ainsi que leurs carrés. On a :

$$\begin{aligned} b_{pp}^2 + b_{qq}^2 &= a_{pp}^2 + a_{qq}^2 + 2t^2 a_{pq}^2 + 2ta_{pq}(a_{qq} - a_{pp}) \\ &= a_{pp}^2 + a_{qq}^2 + 2a_{pq}^2 + 2a_{pq}(t^2 a_{pq} + t(a_{qq} - a_{pp}) - a_{pq}). \end{aligned}$$

Or, le choix de t fait que l’on a $t^2 + t \frac{a_{qq} - a_{pp}}{a_{pq}} - 1 = 0$.

D’où le résultat énoncé : $b_{pp}^2 + b_{qq}^2 = a_{pp}^2 + a_{qq}^2 + 2t^2 a_{pq}^2$. \diamond

Ce théorème montre que le poids de la matrice se déporte, au cours des itérations de la méthode de Jacobi, sur la diagonale de la matrice et, par conséquent, que les éléments non diagonaux, eux, ont un poids qui diminue. Par ailleurs, il semble que pour accélérer la convergence du procédé, on ait intérêt à choisir comme couple (p, q) les indices d’un élément non diagonal de module maximum. C’est effectivement ce choix qui est fait dans la méthode de Jacobi dite *classique*.

4.1. MÉTHODE DE JACOBI

37

Théorème 4.2. La suite des matrices obtenues par la méthode de Jacobi classique est convergente et converge vers une matrice diagonale contenant les valeurs propres de A .

La méthode de Jacobi classique permet aussi d’obtenir une approximation des vecteurs propres d’une matrice A , au moins quand les valeurs propres de A sont distinctes. C’est ce que précise le théorème suivant, admis sans démonstration.

Théorème 4.3. Si la méthode de Jacobi classique converge en un nombre fini d’itérations ou si, en cas de convergence en un nombre infini d’itérations, toutes les valeurs propres de la matrice A sont distinctes, alors la suite des produits des matrices Ω (en mettant à chaque étape la nouvelle matrice Ω à droite du produit) converge vers une matrice orthogonale dont les vecteurs colonnes constituent un ensemble orthonormal de vecteurs propres de la matrice A .

Exemple 1. Appliquons la méthode de Jacobi à la recherche d’approximations

des valeurs propres et des vecteurs propres de la matrice $A = \begin{pmatrix} 1 & 2 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 5 \end{pmatrix}$.

Il n’y a que les coefficients $p = 1$ et $q = 2$ qui sont à considérer. Avec les notations précédentes, on a $x = 0$ et donc $t = 1$, $s = c = \frac{\sqrt{2}}{2}$. Par conséquent,

la matrice Ω vaut $\Omega = \begin{pmatrix} \sqrt{2}/2 & \sqrt{2}/2 & 0 \\ -\sqrt{2}/2 & \sqrt{2}/2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$.

L’application des formules précédentes donne :

- terme inchangé (ici, un seul *a priori*) : $b_{33} = a_{33} = 5$

- première ligne et première colonne, sauf diagonale :

$$b_{12} = b_{21} = 0$$

$$b_{13} = b_{31} = ca_{13} - sa_{23} = 0$$

- deuxième ligne et deuxième colonne, sauf diagonale :

$$b_{23} = b_{32} = sa_{13} + ca_{23} = 0$$

- termes diagonaux qui changent *a priori* :

$$b_{11} = a_{11} - ta_{12} = 1 - 2 = -1$$

$$b_{22} = a_{22} + ta_{12} = 1 + 2 = 3.$$

On obtient donc : $B = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix}.$

La matrice est diagonale, la méthode de Jacobi converge ici en une itération (l'exemple est très simple) et nous donne les valeurs propres (exactement) ainsi que les vecteurs propres de A . Les valeurs propres de A valent : -1 , 3 et 5 . La base orthonormale de vecteurs propres est constituée des vecteurs : $(\sqrt{2}/2, -\sqrt{2}/2, 0)^t$, $(\sqrt{2}/2, \sqrt{2}/2, 0)^t$, $(0, 0, 1)^t$.

Exemple 2. Appliquons la méthode de Jacobi à la recherche d'approximations des valeurs propres et des vecteurs propres de la matrice $A = \begin{pmatrix} 1 & 2 & 4 \\ 2 & -3 & -1 \\ 4 & -1 & 7 \end{pmatrix}.$

Première étape.

Choisissons la plus grande valeur absolue d'un coefficient non diagonal : il s'agit de la valeur 4 , avec $p = 1, q = 3$.

On calcule $x : x = \frac{7-1}{2 \times 4} = \frac{3}{4}.$

On résout l'équation $t^2 + 2xt - 1 = 0$, c'est-à-dire : $t^2 + \frac{3}{2}t - 1 = 0$, qui a pour racines $t = 1/2$ et $t = -2$. On retient la plus petite racine en valeur absolue : $t = 1/2$.

On calcule c et $s : c = \frac{1}{\sqrt{1+t^2}} = \frac{2}{\sqrt{5}} = \frac{2\sqrt{5}}{5}$ et $s = tc = \frac{\sqrt{5}}{5}.$

Puis on applique les formules donnant les coefficients de B , avec bien sûr $b_{13} = b_{31} = 0 :$

b_{22} reste inchangé : $b_{22} = -3$

$$b_{12} = b_{21} = ca_{12} - sa_{32} = \frac{2\sqrt{5}}{5} \times 2 - \frac{\sqrt{5}}{5} \times (-1) = \sqrt{5}$$

$$b_{32} = b_{23} = sa_{12} + ca_{32} = \frac{\sqrt{5}}{5} \times 2 + \frac{2\sqrt{5}}{5} \times (-1) = 0$$

$$b_{11} = a_{11} - ta_{13} = 1 - \frac{1}{2} \times 4 = -1$$

$$b_{33} = a_{33} + ta_{13} = 7 + \frac{1}{2} \times 4 = 9.$$

On obtient ainsi la matrice $B :$

$$B = \begin{pmatrix} -1 & \sqrt{5} & 0 \\ \sqrt{5} & -3 & 0 \\ 0 & 0 & 9 \end{pmatrix}$$

4.1. MÉTHODE DE JACOBI

39

et la matrice de passage Ω_1 :

$$\Omega_1 = \begin{pmatrix} 2\frac{\sqrt{5}}{5} & 0 & \frac{\sqrt{5}}{5} \\ 0 & 1 & 0 \\ -\frac{\sqrt{5}}{5} & 0 & 2\frac{\sqrt{5}}{5} \end{pmatrix} \approx \begin{pmatrix} 0,894 & 0 & 0,447 \\ 0 & 1 & 0 \\ -0,447 & 0 & 0,894 \end{pmatrix}.$$

Seconde étape. On repart de la matrice B pour passer à une matrice C calculée avec la méthode de Jacobi.

On pose : $p = 1, q = 2$.

On calcule x : $x = \frac{-3+1}{2\sqrt{5}} = -\frac{\sqrt{5}}{5}$.

On résout l'équation : $t^2 - 2\frac{\sqrt{5}}{5}t - 1 = 0$ qui a pour racines : $t = \frac{\sqrt{5}}{5}(1 + \sqrt{6})$ et $t = \frac{\sqrt{5}}{5}(1 - \sqrt{6})$. On retient la plus petite racine en valeur absolue : $t = \frac{\sqrt{5}}{5}(1 - \sqrt{6}) \approx -0,648$. En conséquence : $c = \frac{1}{\sqrt{1+t^2}} \approx 0,839$ et $s = ct \approx -0,544$. On obtient alors :

$$c_{33} = b_{33} = 9$$

$$c_{12} = c_{21} = 0$$

$$c_{11} = b_{11} - tb_{12} = -1 - \frac{\sqrt{5}}{5}(1 - \sqrt{6})\sqrt{5} = -2 + \sqrt{6}$$

$$c_{22} = b_{22} + tb_{12} = -3 + \frac{\sqrt{5}}{5}(1 - \sqrt{6})\sqrt{5} = -2 - \sqrt{6}$$

$$c_{13} = c_{31} = cb_{13} - sb_{23} = 0$$

$$c_{23} = c_{32} = sb_{13} + cb_{23} = 0.$$

D'où C :

$$C = \begin{pmatrix} -2 + \sqrt{6} & 0 & 0 \\ 0 & -2 - \sqrt{6} & 0 \\ 0 & 0 & 9 \end{pmatrix}.$$

La matrice de passage Ω_2 approchée est donnée par :

$$\Omega_2 \approx \begin{pmatrix} 0,839 & -0,544 & 0 \\ 0,544 & 0,839 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

La matrice C étant diagonale, la méthode est terminée. Les valeurs propres de A valent : $-2 + \sqrt{6}, -2 - \sqrt{6}, 9$.

Une base orthonormale approchée de vecteurs propres s'obtient en calculant le produit $\Omega_1\Omega_2$: $\Omega_1\Omega_2 \approx \begin{pmatrix} 0,75 & -0,486 & 0,447 \\ 0,544 & 0,839 & 0 \\ -0,375 & 0,243 & 0,894 \end{pmatrix}.$

Exemple 3. Appliquons la méthode de Jacobi à la recherche d’approximations des valeurs propres et des vecteurs propres de la matrice $A = \begin{pmatrix} 1 & 2 & 4 \\ 2 & -3 & 0 \\ 4 & 0 & 7 \end{pmatrix}$.

Nous choisissons la plus grande valeur absolue d’un coefficient non diagonal. Il s’agit de la valeur 4. On pose : $p = 1, q = 3$.

On calcule comme dans l’exemple 2 : $t = 1/2$, $c = \frac{2\sqrt{5}}{5}$ et $s = \frac{\sqrt{5}}{5}$.

On a :

$$b_{22} = -3$$

$$b_{13} = b_{31} = 0$$

$$b_{12} = b_{21} = ca_{12} - sa_{32} = \frac{2\sqrt{5}}{5} \times 2 - \frac{\sqrt{5}}{5} \times 0 = \frac{4\sqrt{5}}{5}$$

$$b_{32} = b_{23} = sa_{12} + ca_{32} = \frac{\sqrt{5}}{5} \times 2 + \frac{2\sqrt{5}}{5} \times 0 = \frac{2\sqrt{5}}{5}$$

$$b_{11} = a_{11} - ta_{13} = -1$$

$$b_{33} = a_{33} + ta_{13} = 9.$$

$$\text{On obtient donc : } B = \begin{pmatrix} -1 & 4\frac{\sqrt{5}}{5} & 0 \\ 4\frac{\sqrt{5}}{5} & -3 & 2\frac{\sqrt{5}}{5} \\ 0 & 2\frac{\sqrt{5}}{5} & 9 \end{pmatrix}.$$

Cet exemple montre que des coefficients peuvent passer de nuls à non nuls. Néanmoins, en passant de A à B , le poids de la matrice s’est concentré sur la diagonale. Il faudrait poursuivre la méthode pour calculer une approximation des valeurs propres et des vecteurs propres de la matrice A .

Deuxième partie

Optimisation linéaire

Chapitre 5

Optimisation linéaire : l’algorithme du simplexe

5.1 Introduction

Afin d’illustrer ce qu’est l’*optimisation linéaire*, aussi appelée *programmation linéaire*, commençons par un exemple simple. Nous pourrions ainsi introduire certaines propriétés des problèmes relevant de ce domaine, propriétés qui seront ensuite exploitées pour fonder l’*algorithme du simplexe*¹. Conçu par G. Dantzig à partir de 1947², il est devenu un des principaux algorithmes d’optimisation linéaire, même si d’autres algorithmes sont venus depuis le concurrencer, notamment la méthode de N. Karmakar³.

1. Le nom de cette méthode peut paraître un peu trompeur. En géométrie, un simplexe de dimension d , ou d -simplexe, est l’enveloppe convexe de $d+1$ points. Ainsi, un 1-simplexe est un segment de droite, un 2-simplexe est un triangle et un 3-simplexe est un tétraèdre. L’algorithme du simplexe ne se limite pas à des simplexes, mais considère plus généralement des polyèdres.

2. G.B. Dantzig, Linear Programming, in *Problems for the Numerical Analysis of the Future, Proceedings of Symposium on Modern Calculating Machinery and Numerical Methods*, UCLA, 1948. Voir aussi G.B. Dantzig, M.N. Thapa, *Linear Programming 1: Introduction*, 1997, et *Linear Programming 2: Theory and Extensions*, 2003, Springer-Verlag et V. Chvátal, *Linear Programming*, 1983, Freeman and company, livre auquel nous empruntons certains exemples.

3. N. Karmarkar, A New Polynomial Time Algorithm for Linear Programming, *Combinatorica* 4 (4), 1984, 373-395.

Une usine fabrique deux sortes de produits, p_1 et p_2 , à l’aide de deux machines m_1 et m_2 . On suppose que la quantité fabriquée de ces produits n’est pas nécessairement un nombre entier, mais seulement un réel positif ou nul. Chaque unité de produit en cours de fabrication doit passer sur les deux machines dans un ordre indifférent et pendant les temps suivants, exprimés en minutes :

	p_1	p_2
m_1	30	20
m_2	40	10

La machine m_1 est disponible 6000 minutes par mois et la machine m_2 est disponible 4000 minutes par mois. Le profit réalisé sur une unité du produit p_1 est de 400 €. Le profit réalisé sur une unité du produit p_2 est de 200 €.

On souhaite trouver le plan de fabrication mensuel qui maximise le profit. Appelons x_1 (respectivement x_2) le nombre d’unités du produit p_1 (respectivement p_2) à fabriquer mensuellement et z le profit réalisé (on suppose que le profit est additif : il n’y a pas d’effet de synergie ou de concurrence entre les produits). Ce problème peut donc s’exprimer sous la forme suivante :

$$\begin{aligned} &\text{Maximiser } z = 400x_1 + 200x_2 \\ &\text{avec les contraintes : } \begin{cases} 30x_1 + 20x_2 \leq 6000 \\ 40x_1 + 10x_2 \leq 4000 \\ x_1 \geq 0, x_2 \geq 0. \end{cases} \end{aligned}$$

Le problème étant à deux variables, il est facile à résoudre graphiquement comme on le voit sur la figure 5.1.

Les points (x_1, x_2) qui satisfont les contraintes appartiennent au quadrilatère $OABC$. Soit λ un réel. La famille :

$$D_\lambda = \{(x_1, x_2) \mid 400x_1 + 200x_2 = \lambda\}$$

est une famille de droites parallèles. Parmi celles de ces droites qui ont une intersection non vide avec le quadrilatère, celle qui passe par B correspond à la plus grande valeur de λ : elle rencontre le quadrilatère des contraintes au point de coordonnées (40, 240). La solution optimale du problème est donc $x_1 = 40$, $x_2 = 240$ (et $z = 64\,000$ €).

Plus généralement, un *problème d’optimisation linéaire* est un problème qui peut se formuler comme suit :

$$\text{maximiser une forme linéaire de } n \text{ variables } x_1, \dots, x_n : \sum_{j=1}^n c_j x_j$$

5.1. INTRODUCTION

45

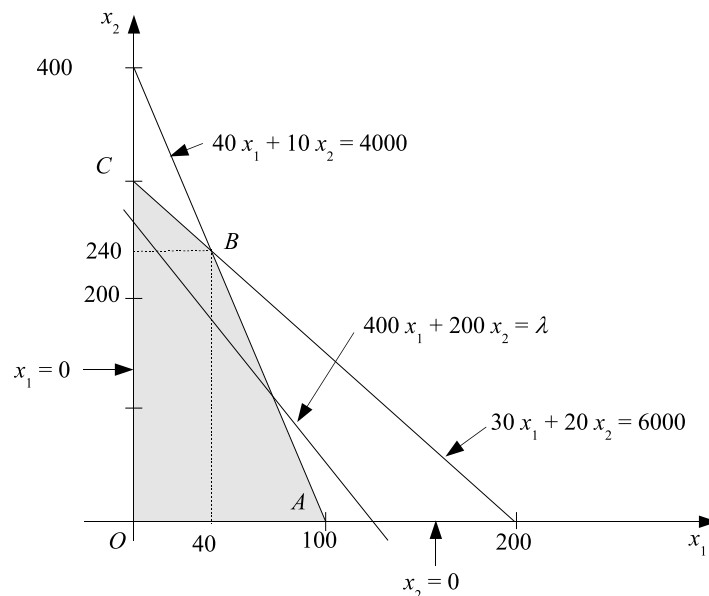


FIGURE 5.1 – Illustration de l'exemple.

les variables étant soumises :

- à m contraintes linéaires : pour $i \in \{1, 2, \dots, m\}$, $\sum_{j=1}^n a_{ij}x_j \leq b_i$
- aux n contraintes de positivité : pour $j \in \{1, 2, \dots, n\}$, $x_j \geq 0$,

où les c_j ($1 \leq j \leq n$), a_{ij} ($1 \leq i \leq m$, $1 \leq j \leq n$) et b_i ($1 \leq i \leq m$) sont des constantes réelles.

Cette formulation s'appelle la *forme standard* d'un problème d'optimisation linéaire. On peut envisager d'autres formulations du problème à résoudre. Dans toute la suite, on ne considérera pas le cas d'inégalités strictes (le domaine défini par les contraintes n'étant plus alors un fermé – voir la définition 8.5., page 143 –, le problème pourrait ne pas admettre de solution optimale, même si la fonction z est majorée sur ce domaine). En revanche, des problèmes où il s'agit de minimisation, ou pour lesquels apparaissent des contraintes d'égalité ou d'inégalité large dans l'autre sens, ou encore pour lesquels certaines variables ont d'autres contraintes que celles d'être positives ou nulles peuvent facilement se mettre sous forme standard, comme le précisent les indications suivantes :

- minimiser une fonction f (linéaire ou non) revient à maximiser $-f$, puisqu'on a la relation : minimum de $f = -$ maximum de $(-f)$;
- on transforme une inégalité du genre « \geq » en une inégalité du genre « \leq » en la multipliant par -1 ;
- une égalité $\sum_{j=1}^n a_{ij}x_j = b_i$ revient aux deux inégalités $\sum_{j=1}^n a_{ij}x_j \leq b_i$ et $\sum_{j=1}^n (-a_{ij})x_j \leq -b_i$;
- on remplace une variable x contrainte par l'inégalité $x \geq \alpha$ par la variable $y = x - \alpha$ qui devra être positive ou nulle (s'il y a plusieurs contraintes de ce type, on ne considère que celle ayant la plus grande valeur α et on élimine les autres) ;
- on remplace une variable x contrainte par l'inégalité $x \leq \beta$ par la variable $y = \beta - x$ qui devra être positive ou nulle (s'il y a plusieurs contraintes de ce type, on ne considère que celle ayant la plus petite valeur β et on élimine les autres) ;
- on remplace une variable contrainte par la double inégalité $\alpha \leq x \leq \beta$ par la variable $y = x - \alpha$ et on ajoute les contraintes $y \leq \beta - \alpha$ et $y \geq 0$ (s'il existe plusieurs contraintes de ce type impliquant une même variable x , on ne garde que l'encadrement le plus contraignant) ;
- on exprime une variable x qui n'est contrainte ni à être positive ni à être négative comme étant la différence de deux variables positives ou nulles : $x = x^+ - x^-$ avec $x^+ \geq 0$ et $x^- \geq 0$.

On peut se demander si la démarche proposée pour l'exemple précédent est susceptible d'être généralisée à la résolution de tout problème d'optimisation linéaire. Puisqu'il est toujours possible d'exprimer un problème d'optimisation linéaire (sans contrainte d'inégalité stricte) sous forme standard, considérons un problème décrit sous cette forme :

$$\begin{aligned} & \text{Maximiser } z = \sum_{j=1}^n c_j x_j \\ & \text{avec les contraintes : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n a_{ij} x_j \leq b_i \\ \text{pour } j \in \{1, 2, \dots, n\}, x_j \geq 0. \end{cases} \end{aligned}$$

L'ensemble des points de \mathbb{R}^n de coordonnées x_1, \dots, x_n vérifiant les $m + n$ contraintes précédentes constitue un polyèdre appelé *polyèdre des contraintes*.

5.2. L’ALGORITHME DU SIMPLEXE SUR UN EXEMPLE

47

Ce polyèdre est *convexe*, c’est-à-dire que, pour tout point M et tout point P du polyèdre, le segment $[M, P]$ est entièrement contenu dans le polyèdre. En effet, soient $M = (x_1, \dots, x_n)$ et $P = (y_1, \dots, y_n)$ deux points quelconques du polyèdre déterminé par les contraintes ; alors, pour tout réel λ vérifiant $0 \leq \lambda \leq 1$, il est facile de vérifier que le point $\lambda M + (1 - \lambda)P$ (de coordonnées $\lambda x_i + (1 - \lambda)y_i$) appartient au polyèdre. Les n -uplets (x_1, \dots, x_n) qui satisfont les contraintes s’appellent *solutions réalisables* du problème. Ce sont les coordonnées des points intérieurs (au sens large) du polyèdre des contraintes qui, dans l’exemple, était le quadrilatère $OABC$.

Le développement de l’algorithme du simplexe montrera le théorème suivant :

Théorème 5.1. Soit un problème d’optimisation linéaire dont le polyèdre des contraintes est non vide et dont la fonction à maximiser est majorée sur ce polyèdre. Alors le problème admet un maximum (fini) atteint en au moins un sommet du polyèdre des contraintes.

L’idée de l’algorithme du simplexe est de passer itérativement d’un sommet du polyèdre des contraintes à un sommet adjacent en suivant des arêtes du polyèdre de façon à augmenter la valeur de la fonction à optimiser, jusqu’à trouver un sommet où le maximum est atteint. C’est grâce à la convexité du polyèdre et à la linéarité de la fonction dont on cherche le maximum que l’on peut se contenter de chercher le maximum en un sommet du polyèdre (notons que, pour certains problèmes, il peut aussi exister des solutions optimales ailleurs qu’en un sommet du polyèdre ; ainsi, dans l’exemple précédent, si z vaut $300x_1 + 200x_2$ au lieu de $400x_1 + 200x_2$, tout le segment $[B, C]$ est constitué de solutions optimales ; cela n’invalide cependant pas ce qui précède).

5.2 L’algorithme du simplexe sur un exemple

Appliquons maintenant l’algorithme du simplexe à un exemple plus sophistiqué, afin d’en illustrer le fonctionnement.

Une fabrique de tissus produit quatre types de tissus : du kelsch, du nanzouk, du shantung et du zénana. Ces tissus résultent de trois opérations principales : la filature, le tissage, la teinture. Ils sont produits en longueur variable, mesurée ici en kilomètres. La production d’un kilomètre de tissu

nécessite un certain nombre d’heures de filature, de tissage et de teinture, ces nombres dépendant du tissu. Par ailleurs, la vente de ces tissus rapporte un certain bénéfice exprimé en euros. Ces données sont précisées dans le tableau suivant pour un kilomètre de tissu :

	kelsch	nanzouk	shantung	zénana
filature	2	4	5	7
tissage	1	1	2	2
teinture	1	2	3	3
bénéfice	7	9	18	17

L’entreprise dispose, quotidiennement, de 42 heures de filature, 17 heures de tissage et 24 heures de teinture. On souhaite établir un plan de fabrication de façon à maximiser le bénéfice (on suppose que l’on est en régime stable de fabrication et non en phase initiale où il faut filer avant de tisser et tisser avant de teindre).

Appelons x_1, x_2, x_3, x_4 les longueurs respectives de kelsch, de nanzouk, de shantung et de zénana produites quotidiennement. Le problème admet alors la modélisation suivante :

$$\begin{aligned} &\text{Maximiser } z = 7x_1 + 9x_2 + 18x_3 + 17x_4 \\ &\text{avec les contraintes : } \begin{cases} 2x_1 + 4x_2 + 5x_3 + 7x_4 \leq 42 \\ x_1 + x_2 + 2x_3 + 2x_4 \leq 17 \\ x_1 + 2x_2 + 3x_3 + 3x_4 \leq 24 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0. \end{cases} \end{aligned}$$

On reconnaît un problème d’optimisation linéaire sous forme standard. On va résoudre ce problème à l’aide de l’algorithme du simplexe que nous expliquerons ainsi sur cet exemple.

On introduit trois variables dites *variables d’écart* x_5, x_6, x_7 , positives ou nulles, qui mesurent pour chaque ressource l’écart entre la quantité initialement disponible et la quantité consommée par le plan de fabrication (caractérisé par x_1, x_2, x_3 et x_4). On obtient ce qui s’appelle un *dictionnaire* (voir plus bas la définition générale), le premier pour la résolution de ce problème (il y en aura d’autres) :

$$\begin{array}{rcll} & x_5 & = & 42 - 2x_1 - 4x_2 - 5x_3 - 7x_4 \\ \text{Dictionnaire I} & x_6 & = & 17 - x_1 - x_2 - 2x_3 - 2x_4 \\ & x_7 & = & 24 - x_1 - 2x_2 - 3x_3 - 3x_4 \\ & z & = & 7x_1 + 9x_2 + 18x_3 + 17x_4 \end{array}$$

Le problème s’écrit maintenant :

5.2. L’ALGORITHME DU SIMPLEXE SUR UN EXEMPLE

49

Maximiser z avec $x_k \geq 0$ pour $1 \leq k \leq 7$.

Le polyèdre des contraintes est limité dans \mathbb{R}^4 par les hyperplans d’équation $x_k = 0$ pour $1 \leq k \leq 7$.

Dans ce dictionnaire, les variables x_5 , x_6 et x_7 sont exprimées comme fonctions affines des variables x_1 , x_2 , x_3 et x_4 ; on traduit cette caractéristique en disant que les variables x_5 , x_6 et x_7 sont actuellement les *variables de base* du dictionnaire et les variables x_1 , x_2 , x_3 et x_4 les *variables hors-base* du dictionnaire (la définition 5.2., page 52, précise plus loin ce qu’est une *base*). On s’intéresse alors à ce qu’on appelle la *solution basique* associée au dictionnaire : c’est la solution obtenue en attribuant la valeur 0 à toutes les variables hors-base ; les valeurs des variables de base en découlent.

Afin de distinguer les fonctions et les variables des valeurs de ces fonctions et de ces variables, on utilisera le signe $*$ lorsqu’il s’agit de valeurs : ainsi x^* représentera une valeur prise par la variable x . Avec cette notation, les égalités $x_1^* = 0$, $x_2^* = 0$, $x_3^* = 0$, $x_4^* = 0$ entraînent $x_5^* = 42$, $x_6^* = 17$ et $x_7^* = 24$. Les sept variables ayant des valeurs positives ou nulles dans cette solution basique, on dit que ce dictionnaire est *réalisable*. On peut remarquer que le point de coordonnées $(0, 0, 0, 0)$ est ici un sommet du polyèdre des contraintes ; la solution basique associée au dictionnaire donne alors à z la valeur 0.

La remarque suivante est à la base de la méthode : on considère l’expression de z dans le dictionnaire courant ; si, dans cette expression, on fait croître à partir de 0 une variable hors-base pourvue d’un coefficient strictement positif (les autres variables hors-base restant nulles), la valeur de z croît. Dans notre exemple, choisissons la variable x_3 (on pourrait aussi choisir ici l’une quelconque des trois autres variables hors-base). Gardant x_1 , x_2 et x_4 à 0, nous cherchons à augmenter x_3 au maximum, tout en conservant la propriété que le point M de \mathbb{R}^4 de coordonnées $(0, 0, x_3, 0)$ reste dans le polyèdre des contraintes (on se déplace alors sur une arête du polyèdre des contraintes issue du sommet $(0, 0, 0, 0)$).

Les contraintes sur l’augmentation de la variable x_3 sont :

$x_5 \geq 0$, ce qui impose $x_3 \leq 8,4$;

$x_6 \geq 0$, ce qui impose $x_3 \leq 8,5$;

$x_7 \geq 0$, ce qui impose $x_3 \leq 8$.

Le premier hyperplan que rencontre le point M est donc celui d’équation $x_7 = 0$: le point M est alors arrivé à un nouveau sommet du polyèdre des contraintes, à l’intersection des hyperplans d’équations $x_1 = 0$, $x_2 =$

0, $x_4 = 0$, $x_7 = 0$. Nous allons alors faire un changement de dictionnaire en échangeant les rôles de x_3 et x_7 pour itérer le procédé que nous venons d'employer. On utilise l'équation du dictionnaire I qui donne x_7 pour exprimer x_3 en fonction de x_1 , x_2 , x_4 et x_7 ; on remplace ensuite x_3 par cette expression dans les autres équations du dictionnaire.

On obtient ainsi un deuxième dictionnaire :

$$\begin{array}{rcll} \text{Dictionnaire II} & x_3 & = & 8 - 1/3 x_1 - 2/3 x_2 - x_4 - 1/3 x_7 \\ & x_5 & = & 2 - 1/3 x_1 - 2/3 x_2 - 2x_4 + 5/3 x_7 \\ & x_6 & = & 1 - 1/3 x_1 + 1/3 x_2 + 2/3 x_7 \\ & z & = & 144 + x_1 - 3x_2 - x_4 - 6x_7 \end{array}$$

On dit qu'on a fait « entrer x_3 en base » et qu'on a fait « sortir x_7 de la base », ou encore que x_3 est la *variable entrante* et que x_7 est la *variable sortante*. Les variables de base sont maintenant x_3 , x_5 et x_6 , et les variables hors-base x_1 , x_2 , x_4 et x_7 . Dans la nouvelle solution basique, la fonction z vaut 144, valeur que l'on obtient en annulant les variables hors-base. On obtient ainsi une nouvelle solution réalisable plus intéressante que celle associée au premier dictionnaire.

Dans la nouvelle expression de la fonction z , nous voyons que seule la variable x_1 est affectée d'un coefficient strictement positif : on fait entrer x_1 en base, et on parcourt ainsi une nouvelle arête du polyèdre des contraintes ; on a les limites suivantes sur l'augmentation possible de la valeur de x_1 à partir de la valeur nulle, les autres variables hors-base restant à 0 :

$$\begin{aligned} x_3 &\geq 0, \text{ ce qui impose } x_1 \leq 24; \\ x_5 &\geq 0, \text{ ce qui impose } x_1 \leq 6; \\ x_6 &\geq 0, \text{ ce qui impose } x_1 \leq 3. \end{aligned}$$

C'est la troisième limite qui est la plus contraignante ; x_6 sort de la base, ce qui conduit au dictionnaire suivant :

$$\begin{array}{rcll} \text{Dictionnaire III} & x_1 & = & 3 + x_2 - 3x_6 + 2x_7 \\ & x_3 & = & 7 - x_2 - x_4 + x_6 - x_7 \\ & x_5 & = & 1 - x_2 - 2x_4 + x_6 + x_7 \\ & z & = & 147 - 2x_2 - x_4 - 3x_6 - 4x_7 \end{array}$$

La solution basique associée à ce nouveau dictionnaire donne à z la valeur 147.

Nous voyons sur la dernière ligne du dictionnaire III que, les variables x_2 , x_4 , x_6 , x_7 étant positives ou nulles, l'optimum cherché de z est majoré

5.3. DÉFINITIONS ET TERMINOLOGIE

51

par 147. La solution basique actuelle nous fournit donc une solution optimale du problème :

- il faut fabriquer chaque jour trois kilomètres de kelsch, zéro de nanzouk, sept de shantung et zéro de zénana ;
- toutes les heures de tissage et de teinture sont utilisées, alors qu’il reste une heure de filage disponible ;
- le bénéfice maximum vaut 147 €.

Remarques.

1. Il se trouve que la solution obtenue ici est entière alors que cela n’était pas imposé par la formulation du problème. Ceci n’a rien de général et les problèmes d’optimisation linéaire en nombres entiers (c’est-à-dire des problèmes d’optimisation linéaire pour lesquels les variables doivent prendre des valeurs entières) peuvent être qualitativement plus difficiles.
2. La méthode consiste, à chaque étape, à faire entrer en base une variable dont le coefficient dans la fonction z à optimiser est strictement positif. Cela ne permet cependant pas toujours d’obtenir une croissance stricte de z . Nous reviendrons sur ce phénomène dans le paragraphe consacré à la « dégénérescence ».
3. Enfin, nous avons eu la chance de trouver, sans difficulté, un sommet du polyèdre des contraintes ou, autrement dit, un dictionnaire réalisable, qui nous a servi de point de départ. En effet, l’origine était réalisable, c’est-à-dire que l’annulation des variables x_1, x_2, \dots, x_n attribue des valeurs positives ou nulles aux variables d’écart (car les b_i étaient tous positifs ou nuls). Nous étudierons plus loin des cas moins favorables.

5.3 Définitions et terminologie

Revenons sur quelques définitions. On considère un problème d’optimisation linéaire mis sous forme standard :

maximiser une forme linéaire z de n variables x_1, \dots, x_n : $z = \sum_{j=1}^n c_j x_j$,

les variables étant soumises :

- à m contraintes linéaires : pour $i \in \{1, 2, \dots, m\}$, $\sum_{j=1}^n a_{ij} x_j \leq b_i$,

- aux n contraintes de positivité : pour $j \in \{1, 2, \dots, n\}, x_j \geq 0$.

Tout n -uplet de valeurs (x_1^*, \dots, x_n^*) satisfaisant les contraintes constitue une *solution réalisable*. Si un problème admet des solutions réalisables, il est dit *réalisable*. Si un problème d’optimisation linéaire n’admet aucune solution réalisable, il est dit *infaisable* ou *non réalisable*.

La fonction z est appelée *fonction objectif*. Les variables x_1, \dots, x_n sont appelées *variables de décision* ou *variables de choix* ou *variables principales* ou encore *variables initiales* ; les variables x_{n+1}, \dots, x_{n+m} s’appellent les *variables d’écart*. Une solution $x_1^*, x_2^*, \dots, x_{n+m}^*$ est *réalisable* si et seulement si toutes ses valeurs sont positives ou nulles ; autrement dit : pour $k \in \{1, 2, \dots, n+m\}, x_k^* \geq 0$. Une solution réalisable qui maximise la fonction objectif est dite *solution optimale*. Si un problème admet des solutions réalisables et que la fonction objectif peut prendre des valeurs arbitrairement grandes, il est dit réalisable *non borné*.

Il y a donc trois types de problèmes :

- les problèmes réalisables et non bornés,
- les problèmes réalisables et bornés,
- les problèmes non réalisables.

Un *dictionnaire* est un système d’équations linéaires liant $x_1, \dots, x_n, x_{n+1}, \dots, x_{n+m}$ et z , et satisfaisant les deux propriétés suivantes :

- les équations constituant le dictionnaire doivent exprimer de manière unique la fonction objectif z et m des $n+m$ variables x_1, \dots, x_{n+m} en fonction des n autres variables ;
- le dictionnaire est équivalent au système définissant les variables d’écart et la fonction objectif, c’est-à-dire à :

$$\begin{aligned} x_{n+1} &= b_1 - \sum_{j=1}^n a_{1j}x_j \\ &\dots \\ x_{n+i} &= b_i - \sum_{j=1}^n a_{ij}x_j \\ &\dots \\ x_{n+m} &= b_m - \sum_{j=1}^n a_{mj}x_j \\ \hline z &= \sum_{j=1}^n c_jx_j \end{aligned}$$

Définition 5.2. Une *base* est un ensemble de m variables (les *variables de base* ou *en base*) qui s’expriment de façon unique et affine en fonction des n

autres variables (les *variables hors-base*), cette expression étant équivalente aux m contraintes d'égalité définissant les m variables d'écart.

Une base définit donc un dictionnaire et réciproquement. Une base étant fixée, on obtient la *solution basique* associée à cette base en attribuant la valeur 0 à toutes les variables hors-base. Géométriquement, une solution à la fois basique et réalisable correspond à un sommet du polyèdre des contraintes.

L'algorithme du simplexe a pour objectif de déterminer une solution optimale parmi les solutions basiques et réalisables (c'est-à-dire parmi les sommets du polyèdre des contraintes).

5.4 Résumé d'une itération

Pour déterminer une telle solution basique réalisable optimale, décrivons une itération de l'algorithme du simplexe de façon générale. Pour cela, définissons deux sous-ensembles formant une partition de l'ensemble des indices : J est l'ensemble des indices des n variables hors-base dans le dictionnaire courant et I celui des m variables de base. Plus précisément :

- $J \subset \{1, 2, \dots, n + m\}$ avec $|J| = n$ (initialement, on a souvent $J = \{1, 2, \dots, n\}$);
- $I = \{1, 2, \dots, n + m\} \setminus J$;
- le dictionnaire courant est décrit par les égalités suivantes (on utilise des symboles « ' » pour distinguer le dictionnaire courant du dictionnaire initial) :

$$\text{pour } i \in I, x_i = b'_i + \sum_{j \in J} a'_{ij} x_j \text{ et } z = z^* + \sum_{j \in J} c'_j x_j ;$$

on suppose que le dictionnaire est réalisable : pour $i \in I$, $b'_i \geq 0$.

L'itération courante se déroule comme suit :

- si tous les coefficients c'_j sont négatifs ou nuls, l'algorithme est terminé : l'annulation des variables hors-base fournit une solution optimale ;
- sinon :
 - ★ on choisit une variable hors-base x_{j_0} pourvue d'un coefficient c'_{j_0} strictement positif dans z ; il s'agit de la variable entrante ; s'il y a plusieurs variables candidates pour entrer en base, on peut par exemple privilégier la variable ayant le plus grand coefficient

dans z (premier critère de Dantzig) ou, ce qui est généralement plus efficace, privilégier la variable entraînant la plus grande augmentation de z (second critère de Dantzig ; voir les exercices 1 et 2) ; on verra un autre choix au paragraphe suivant, en cas de « dégénérescence » ;

- ★ on détermine la variable sortante x_{i_0} comme étant la variable de base qui restreint le plus la croissance de x_{j_0} ; pour cela, on considère, pour $i \in I$ avec $a'_{ij_0} < 0$, les rapports $-b'_i/a'_{ij_0}$ et i_0 est l'indice pour lequel ce rapport est le plus petit (s'il y a plusieurs variables candidates pour sortir de la base, on peut en choisir une arbitrairement ; on verra un choix systématique au paragraphe suivant, là encore en cas de « dégénérescence ») ;
- ★ on extrait x_{j_0} de l'expression courante de x_{i_0} ;
- ★ on remplace x_{j_0} par sa nouvelle expression dans z et dans l'expression des autres variables de base ; on obtient ainsi le nouveau dictionnaire courant à partir duquel on applique l'itération suivante.

Remarques.

1. Quand on passe du dictionnaire courant au dictionnaire suivant, on est sûr que celui-ci est réalisable, par le choix de la variable sortante. Autrement dit, on passe d'une solution basique réalisable à une autre solution basique réalisable, ou encore, d'un sommet du polyèdre des contraintes à un autre sommet de ce polyèdre. Il est donc inutile de vérifier cette propriété quand on obtient le nouveau dictionnaire.
2. Une variable qui entre en base peut en sortir à l'itération suivante. En revanche, une variable qui sort de la base ne peut pas y entrer à l'itération suivante (mais elle peut entrer en base ultérieurement).

5.5 La dégénérescence et le cyclage

Définition 5.3. Une *solution basique réalisable* avec une ou plusieurs variables de base nulles est dite *dégénérée*. Une *base* dont la solution basique associée est dégénérée est dite *dégénérée*.

Exemple.

Considérons le dictionnaire (non dégénéré) suivant :

5.5. LA DÉGÉNÉRESCENCE ET LE CYCLAGE

55

$$\begin{array}{rcl} x_4 & = & 1 \qquad \qquad \qquad - 2x_3 \\ x_5 & = & 3 - 2x_1 + 4x_2 - 6x_3 \\ x_6 & = & 2 + x_1 - 3x_2 - 4x_3 \\ \hline z & = & 2x_1 - x_2 + 8x_3 \end{array}$$

Choisissant de faire entrer x_3 en base, nous voyons que les relations $x_4 \geq 0$, $x_5 \geq 0$, $x_6 \geq 0$ imposent toutes les trois 0,5 comme limite à la croissance de x_3 . Chacune des trois variables x_4, x_5, x_6 est donc candidate à quitter la base. Si nous choisissons x_4 , nous obtenons comme nouveau dictionnaire :

$$\begin{array}{rcl} x_3 & = & 0,5 \qquad \qquad \qquad - 0,5x_4 \\ x_5 & = & \qquad - 2x_1 + 4x_2 + 3x_4 \\ x_6 & = & \qquad x_1 - 3x_2 + 2x_4 \\ \hline z & = & 4 + 2x_1 - x_2 - 4x_4 \end{array}$$

Dans la solution basique associée à ce dictionnaire, x_5 et x_6 prennent une valeur nulle. Du fait de la nullité d’au moins une des variables en base, cette solution basique est dégénérée.

Si nous faisons une itération à partir de ce dictionnaire, nous voyons que, faisant entrer x_1 en base (seule variable à avoir un coefficient positif dans z), la relation $x_5 \geq 0$ impose $x_1 \leq 0$. La plus grande valeur attribuable à x_1 vaut 0 et la valeur z^* n’augmentera donc pas au cours de cette itération.

L’inconvénient de ces inévitables itérations dégénérées est qu’elles peuvent induire un phénomène désastreux pour la convergence de l’algorithme : le *cyclage*.

Définition 5.4. On dit qu’il y a *cyclage* lorsque, à l’issue d’un nombre fini d’itérations, on retrouve un dictionnaire déjà rencontré.

En fait, à cause de l’indépendance des variables hors-base, on retrouve un dictionnaire déjà rencontré dès qu’on retrouve une même partition des $m + n$ variables en variables de base et variables hors-base (cette situation est illustrée par l’exercice 3).

Remarque. Considérons une itération consistant à passer d’un dictionnaire D_1 à un dictionnaire D_2 avec une variable entrante x . On suppose que la valeur de la fonction z dans la solution basique associée à D_2 est la même que dans la solution basique associée à D_1 . Cela n’est possible que si x a la valeur nulle dans les solutions basiques associées à D_1 et D_2 . En conséquence,

aucune valeur des variables ne change pendant l'itération. Si, pendant une suite de dictionnaires, la valeur de la fonction z ne croît pas, aucune variable ne change ; géométriquement, on reste en un même sommet du polyèdre, les déplacements dans les directions envisagées sont en fait d'amplitude nulle.

On peut toujours éviter le cyclage en appliquant la règle du plus petit indice (*règle de Bland*⁴) : lorsqu'on a un choix sur la variable entrante ou sur la variable sortante, on choisit toujours celle de plus petit indice parmi les variables candidates. Nous allons prouver l'efficacité de cette règle.

Théorème 5.5. (Théorème de Bland) Il ne peut y avoir cyclage quand, lors de toute itération effectuée à partir d'un dictionnaire dégénéré, on choisit les variables entrante et sortante comme celles de plus petit indice parmi les variables candidates.

Preuve. Supposons que, appliquant la règle de Bland, l'on retrouve deux fois le même dictionnaire D_0 à l'issue d'une suite d'itérations ayant construit les dictionnaires $D_0, D_1, \dots, D_k = D_0$; tous ces dictionnaires sont nécessairement dégénérés. On appelle *variable versatile* une variable qui, au cours de ces itérations, est tantôt en base, tantôt hors-base (on notera qu'il existe nécessairement des variables versatiles quand il y a cyclage) ; soit t le plus grand indice des variables versatiles. Dans la suite de dictionnaires D_0, D_1, \dots, D_k , D_1, \dots, D_k , il existe nécessairement un dictionnaire D' dans lequel x_t est sortante (c'est-à-dire qu'elle est de base dans D' et pas dans le dictionnaire suivant), puis un dictionnaire D'' où x_t est entrante ; soit x_s la variable qui entre en base lorsque, à partir de D' , x_t sort (x_s n'est pas en base dans D' mais l'est dans le dictionnaire suivant) ; x_s est versatile et on a donc $s < t$.

En notant I l'ensemble des indices des variables de base de D' , on peut écrire D' sous la forme :

$$\begin{aligned} \forall i \in I, x_i &= b'_i - \sum_{j \notin I} a'_{ij} x_j \\ z &= z^* + \sum_{j \notin I} c'_j x_j \end{aligned}$$

4. R.G. Bland, New finite pivoting rules for the simplex method, *Mathematics of Operations Research* 2, 1977, 103-107.

5.5. LA DÉGÉNÉRESCENCE ET LE CYCLAGE

57

La variable x_s étant entrante, on a $c'_s > 0$. La règle de Bland étant utilisée, on a, pour $j \notin I$ avec $j < s$, $c'_j \leq 0$. La variable x_t étant sortante dans D' , il vient $a'_{ts} > 0$.

La dernière ligne de D'' peut quant à elle s'écrire :

$$z = z^* + \sum_{k=1}^{n+m} c''_k x_k$$

où c''_k est nul si x_k est en base et $c''_t > 0$.

Pour toute solution $(x_1^*, \dots, x_{n+m}^*)$ du système des contraintes, on a, puisque la valeur de z^* ne change pas pendant le cyclage :

$$z^* + \sum_{j \notin I} c'_j x_j^* = z^* + \sum_{k=1}^{n+m} c''_k x_k^*.$$

Si on définit une solution particulière du système des contraintes en donnant une valeur nulle à toutes les variables hors-base dans D' sauf à x_s et une valeur quelconque x_s^* à x_s (les valeurs des autres variables sont alors entièrement déterminées), l'égalité ci-dessus devient :

$$c'_s x_s^* = c''_s x_s^* + \sum_{i \in I} c''_i (b'_i - a'_{is} x_s^*)$$

ou encore :

$$\left(c'_s - c''_s + \sum_{i \in I} c''_i a'_{is} \right) x_s^* = \sum_{i \in I} c''_i b'_i.$$

Cette égalité étant vraie pour toute valeur x_s^* , il vient :

$$c'_s - c''_s + \sum_{i \in I} c''_i a'_{is} = 0.$$

Puisque c'est x_t qui est entrante dans D'' et non x_s alors que l'on a $s < t$, c'est que nous avons $c''_s \leq 0$. Comme nous avons remarqué l'inégalité $c'_s > 0$, il existe un indice r de I avec $c''_r a'_{rs} < 0$.

Par définition de r , la variable x_r était en base dans D' et puisque c''_r est non nul, elle n'est pas en base dans D'' . Nous en déduisons que x_r est une variable versatile, d'où l'inégalité $r \leq t$.

De plus, c''_t et a'_{ts} étant positifs, leur produit l'est aussi et r ne peut donc pas être égal à t , d'où $r < t$.

Comme x_t entre en base dans D'' alors qu'on a $r < t$, c'est que x_r n'est pas entrante dans D'' et nous n'avons donc pas $c''_r > 0$; par conséquent c'est que nous avons $a'_{rs} > 0$.

D'après la remarque faite plus haut, toutes les variables versatiles gardent la valeur nulle au cours du cyclage. La variable x_r étant versatile, elle est nulle dans la solution basique associée à D' . En conséquence, on a $b'_r = 0$.

La variable x_r était donc candidate à quitter la base de D' au même titre que x_t ; en choisissant x_t avec $t > r$, nous n'avons pas appliqué la règle du plus petit indice, contradiction. \diamond

Remarque. Il est inutile d'appliquer la règle de Bland lorsque le dictionnaire n'est pas dégénéré.

5.6 Recherche d'un dictionnaire réalisable

Nous allons ici encore nous appuyer sur un exemple. Supposons que nous voulions résoudre le problème suivant, écrit sous forme standard.

$$\begin{aligned} &\text{Maximiser } z = x_1 - x_2 + x_3 \\ &\text{avec les contraintes : } \begin{cases} 2x_1 - x_2 + 2x_3 \leq 4 \\ 2x_1 - 3x_2 + x_3 \leq -5 \\ -x_1 + x_2 - 2x_3 \leq -1 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0. \end{cases} \end{aligned}$$

Nous introduisons le *problème auxiliaire* suivant.

$$\begin{aligned} &\text{Minimiser } x_0 \\ &\text{avec les contraintes : } \begin{cases} 2x_1 - x_2 + 2x_3 \leq 4 + x_0 \\ 2x_1 - 3x_2 + x_3 \leq -5 + x_0 \\ -x_1 + x_2 - 2x_3 \leq -1 + x_0 \\ x_0 \geq 0, x_1 \geq 0, x_2 \geq 0, x_3 \geq 0. \end{cases} \end{aligned}$$

D'une façon plus générale, on obtient le problème auxiliaire en ajoutant x_0 aux b_i et en minimisant x_0 . Si le problème initial est le suivant :

$$\begin{aligned} & \text{Maximiser } z = \sum_{j=1}^n c_j x_j \\ & \text{avec les contraintes : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n a_{ij} x_j \leq b_i \\ \text{pour } j \in \{1, 2, \dots, n\}, x_j \geq 0 \end{cases} \end{aligned}$$

alors le problème auxiliaire a pour expression :

$$\begin{aligned} & \text{Minimiser } x_0 \\ & \text{avec les contraintes : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n a_{ij} x_j \leq b_i + x_0 \\ \text{pour } j \in \{0, 1, 2, \dots, n\}, x_j \geq 0. \end{cases} \end{aligned}$$

On peut interpréter ceci en considérant que l'on augmente les ressources d'une quantité x_0 . Il est évident que si x_0 est assez grand, les nouvelles ressources deviennent toutes positives ou nulles. Le problème auxiliaire est donc toujours réalisable.

On établit facilement la proposition suivante :

Proposition 5.6. Le problème initial est réalisable si et seulement si le problème auxiliaire admet 0 pour valeur optimale de la fonction objectif.

En outre, si le problème auxiliaire admet 0 comme valeur optimale, toute solution optimale du problème auxiliaire donne une solution réalisable du problème initial, en « oubliant » x_0 (qui vaut 0 dans ce cas).

Remarque. On peut se contenter d'ajouter x_0 aux seconds membres des inégalités correspondant à une valeur négative des seconds membres, comme il est fait dans le corrigé de l'exercice 4.

Revenons à l'exemple et, après avoir mis le problème sous forme standard, écrivons le dictionnaire définissant les variables d'écart du problème auxiliaire :

$$\begin{array}{rcllclclcl} x_4 & = & 4 & - & 2x_1 & + & x_2 & - & 2x_3 & + & x_0 \\ x_5 & = & -5 & - & 2x_1 & + & 3x_2 & - & x_3 & + & x_0 \\ x_6 & = & -1 & + & x_1 & - & x_2 & + & 2x_3 & + & x_0 \\ \hline w & = & & & & & & & & & -x_0 \end{array}$$

Ce dictionnaire n'est pas réalisable puisqu'en donnant la valeur 0 aux variables hors-base x_1, x_2, x_3, x_0 , les variables d'écart x_5 et x_6 prennent des valeurs négatives. Cependant on peut se ramener en une itération à un dictionnaire réalisable. Il suffit de faire entrer x_0 en base et de faire sortir de la base la variable qui est « la plus négative » (ici x_5). On obtient :

$$\begin{array}{rcllclcl} x_0 & = & 5 & + & 2x_1 & - & 3x_2 & + & x_3 & + & x_5 \\ x_4 & = & 9 & & & - & 2x_2 & - & x_3 & + & x_5 \\ x_6 & = & 4 & + & 3x_1 & - & 4x_2 & + & 3x_3 & + & x_5 \\ \hline w & = & -5 & - & 2x_1 & + & 3x_2 & - & x_3 & - & x_5 \end{array}$$

Dans ce dictionnaire, x_2 est variable entrante. Déterminons la variable sortante :

$$x_0 \geq 0 \text{ implique } x_2 \leq \frac{5}{3};$$

$$x_4 \geq 0 \text{ implique } x_2 \leq \frac{9}{2};$$

$$x_6 \geq 0 \text{ implique } x_2 \leq 1.$$

C'est x_6 qui quitte la base. Le nouveau dictionnaire est alors :

$$\begin{array}{rcllclclcl} x_2 & = & 1 & + & 0,75x_1 & + & 0,75x_3 & + & 0,25x_5 & - & 0,25x_6 \\ x_0 & = & 2 & - & 0,25x_1 & - & 1,25x_3 & + & 0,25x_5 & + & 0,75x_6 \\ x_4 & = & 7 & - & 1,5x_1 & - & 2,5x_3 & + & 0,5x_5 & + & 0,5x_6 \\ \hline w & = & -2 & + & 0,25x_1 & + & 1,25x_3 & - & 0,25x_5 & - & 0,75x_6 \end{array}$$

À l'étape suivante, faisons entrer x_3 en base : x_0 en sort pour donner le dernier dictionnaire du problème auxiliaire.

$$\begin{array}{rcllclclcl} x_3 & = & 1,6 & - & 0,2x_1 & + & 0,2x_5 & + & 0,6x_6 & - & 0,8x_0 \\ x_2 & = & 2,2 & + & 0,6x_1 & + & 0,4x_5 & + & 0,2x_6 & - & 0,6x_0 \\ x_4 & = & 3 & - & x_1 & & & - & x_6 & + & 2x_0 \\ \hline w & = & & & & & & & & - & x_0 \end{array}$$

On voit que le problème initial avait une solution réalisable donnée par $x_1^* = 0$; $x_2^* = 2,2$; $x_3^* = 1,6$. Comme indiqué plus haut, à cause de l'équivalence entre dictionnaires, on déduit un dictionnaire réalisable pour le problème initial en « oubliant » x_0 et en choisissant comme variables en base x_3, x_2, x_4

5.7. EXERCICES

61

exprimées ci-dessus en fonction de x_1, x_5, x_6 . Il suffit d'exprimer z en fonction des mêmes variables. On obtient pour le problème initial le dictionnaire :

$$\begin{array}{rcllclcl} x_3 & = & 1,6 & - & 0,2x_1 & + & 0,2x_5 & + & 0,6x_6 \\ x_2 & = & 2,2 & + & 0,6x_1 & + & 0,4x_5 & + & 0,2x_6 \\ x_4 & = & 3 & - & x_1 & & & - & x_6 \\ \hline z & = & -0,6 & + & 0,2x_1 & - & 0,2x_5 & + & 0,4x_6 \end{array}$$

On peut maintenant partir de ce dictionnaire, réalisable, pour déterminer le maximum de z en appliquant une nouvelle fois l'algorithme du simplexe. Cette méthode est connue sous le nom de *méthode à deux phases*. Dans le chapitre 6, nous verrons que pour certains problèmes où l'origine n'est pas réalisable (parce que certains des b_i sont négatifs), lorsque tous les coefficients c_j sont négatifs, on peut utiliser le problème appelé « dual », ce qui permet de ne résoudre qu'un problème au lieu de deux. Un tel problème est dit *dual-réalisable*.

5.7 Exercices

Exercice 1

Énoncé. Résoudre le problème suivant par l'algorithme du simplexe :

$$\begin{array}{l} \text{Maximiser } z = 3x_1 + 2x_2 + 4x_3 \\ \text{avec les contraintes : } \left\{ \begin{array}{l} x_1 + x_2 + 2x_3 \leq 4 \\ 2x_1 + 3x_3 \leq 5 \\ 2x_1 + x_2 + 3x_3 \leq 7 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0. \end{array} \right. \end{array}$$

Corrigé. Introduisons les variables d'écart du problème. On obtient comme premier dictionnaire :

$$\begin{array}{rcllclcl} x_4 & = & 4 & - & x_1 & - & x_2 & - & 2x_3 \\ x_5 & = & 5 & - & 2x_1 & & & - & 3x_3 \\ x_6 & = & 7 & - & 2x_1 & - & x_2 & - & 3x_3 \\ \hline z & = & & & 3x_1 & + & 2x_2 & + & 4x_3 \end{array}$$

Chacune des trois variables hors-base étant candidate à entrer en base, cherchons celle dont la croissance à partir de 0 permet d'augmenter le plus

la valeur de la fonction objectif, actuellement égale à 0 (second critère de Dantzig). Si x_1 entre en base, comme son augmentation est bornée par $5/2$, la fonction objectif augmente de $15/2$. Si x_2 entre en base, la fonction objectif augmente de 8. Enfin si c'est x_3 , l'objectif augmente de $20/3$. On choisit donc de faire entrer x_2 . La variable en base x_4 contraint le plus l'accroissement de x_2 ; elle quitte la base. On obtient le nouveau dictionnaire :

$$\begin{array}{rclclcl} x_2 & = & 4 & - & x_1 & - & 2x_3 & - & x_4 \\ x_5 & = & 5 & - & 2x_1 & - & 3x_3 & & \\ x_6 & = & 3 & - & x_1 & - & x_3 & + & x_4 \\ \hline z & = & 8 & + & x_1 & & & - & 2x_4 \end{array}$$

Cette fois, nous n'avons plus le choix de la variable entrante, puisque seule x_1 a un coefficient positif dans z , et x_5 quitte la base. Le nouveau dictionnaire est le suivant :

$$\begin{array}{rclclcl} x_1 & = & 5/2 & - & 3/2 x_3 & & - & 1/2 x_5 \\ x_2 & = & 3/2 & - & 1/2 x_3 & - & x_4 & + & 1/2 x_5 \\ x_6 & = & 1/2 & + & 1/2 x_3 & + & x_4 & + & 1/2 x_5 \\ \hline z & = & 21/2 & - & 3/2 x_3 & - & 2x_4 & - & 1/2 x_5 \end{array}$$

Ce dictionnaire est le dernier puisqu'il n'existe plus de variable hors-base dont le coefficient dans z soit strictement positif. Le maximum cherché pour z est donc de $21/2$ et il est obtenu pour les valeurs suivantes des variables :

$$x_1^* = 5/2 ; x_2^* = 3/2 ; x_3^* = 0.$$

Exercice 2

Énoncé. Résoudre le problème suivant par l'algorithme du simplexe :

Q1. en faisant entrer en base la variable de plus grand coefficient dans la fonction objectif (premier critère de Dantzig) ;

Q2. en faisant entrer en base la variable dont l'augmentation permettra d'augmenter le plus la fonction objectif (second critère de Dantzig).

$$\begin{array}{l} \text{Maximiser } z = 5x_1 + 6x_2 + 9x_3 + 8x_4 \\ \text{avec les contraintes : } \left\{ \begin{array}{l} x_1 + 2x_2 + 3x_3 + x_4 \leq 5 \\ x_1 + x_2 + 2x_3 + 3x_4 \leq 3 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0. \end{array} \right. \end{array}$$

5.7. EXERCICES

63

Corrigé. Introduisons les variables d’écart du problème. On obtient comme premier dictionnaire :

$$\begin{array}{rcllclclcl} x_5 & = & 5 & - & x_1 & - & 2x_2 & - & 3x_3 & - & x_4 \\ x_6 & = & 3 & - & x_1 & - & x_2 & - & 2x_3 & - & 3x_4 \\ \hline z & = & & & 5x_1 & + & 6x_2 & + & 9x_3 & + & 8x_4 \end{array}$$

Q1. D’après le critère retenu ici pour faire entrer une variable en base, c’est tout d’abord la variable x_3 qui entre en base. La variable sortante est x_6 . Le nouveau dictionnaire est le suivant :

$$\begin{array}{rcllclclcl} x_3 & = & 1,5 & - & 0,5x_1 & - & 0,5x_2 & - & 1,5x_4 & - & 0,5x_6 \\ x_5 & = & 0,5 & + & 0,5x_1 & - & 0,5x_2 & + & 3,5x_4 & + & 1,5x_6 \\ \hline z & = & 13,5 & + & 0,5x_1 & + & 1,5x_2 & - & 5,5x_4 & - & 4,5x_6 \end{array}$$

Si on choisit encore la variable entrante de plus grand coefficient, il s’agit de x_2 . La variable sortante est alors x_5 . On obtient le dictionnaire ci-dessous :

$$\begin{array}{rcllclclcl} x_2 & = & 1 & + & x_1 & + & 7x_4 & - & 2x_5 & + & 3x_6 \\ x_3 & = & 1 & - & x_1 & - & 5x_4 & + & x_5 & - & 2x_6 \\ \hline z & = & 15 & + & 2x_1 & + & 5x_4 & - & 3x_5 \end{array}$$

La variable x_4 entre maintenant en base et la variable x_3 en sort. D’où :

$$\begin{array}{rcllclclcl} x_4 & = & 0,2 & - & 0,2x_1 & - & 0,2x_3 & + & 0,2x_5 & - & 0,4x_6 \\ x_2 & = & 2,4 & - & 0,4x_1 & - & 1,4x_3 & - & 0,6x_5 & + & 0,2x_6 \\ \hline z & = & 16 & + & x_1 & - & x_3 & - & 2x_5 & - & 2x_6 \end{array}$$

Enfin, la variable x_1 entre en base et x_4 en sort. Le dernier dictionnaire est :

$$\begin{array}{rcllclclcl} x_1 & = & 1 & - & x_3 & - & 5x_4 & + & x_5 & - & 2x_6 \\ x_2 & = & 2 & - & x_3 & + & 2x_4 & - & x_5 & + & x_6 \\ \hline z & = & 17 & - & 2x_3 & - & 5x_4 & - & x_5 & - & 4x_6 \end{array}$$

Tous les coefficients de z sont négatifs ou nuls : la base $\{x_1, x_2\}$ est donc optimale, avec $x_1^* = 1$ et $x_2^* = 2$.

Q2. Envisageons maintenant, à l’aide du tableau ci-dessous, les quatre possibilités pour le choix de la variable entrante :

variable entrante	x_1	x_2	x_3	x_4
accroissement maximum de la variable	3	2,5	1,5	1
accroissement correspondant de z	15	15	13,5	8

Le critère actuel conduit à choisir x_1 ou x_2 . Faisons par exemple entrer x_1 (la conclusion restera la même si on choisit x_2 ici) ; c’est alors la variable x_6 qui sort ; le nouveau dictionnaire est :

$$\begin{array}{rcl}
 x_1 & = & 3 - x_2 - 2x_3 - 3x_4 - x_6 \\
 x_5 & = & 2 - x_2 - x_3 + 2x_4 + x_6 \\
 \hline
 z & = & 15 + x_2 - x_3 - 7x_4 - 5x_6
 \end{array}$$

Seule la variable x_2 est candidate à entrer en base, la variable x_5 sort ; on obtient le même dictionnaire que ci-dessus avec la même conclusion.

Nous remarquons que, avec la première stratégie sur le choix de la variable entrante, le nombre d’étapes vaut quatre alors qu’avec la seconde stratégie, ce nombre vaut deux. Sur ce cas particulier, la seconde stratégie est plus avantageuse.

Exercice 3

Énoncé. On veut appliquer l’algorithme du simplexe au dictionnaire ci-dessous⁵. On envisage deux stratégies lorsqu’il y a plusieurs variables candidates pour entrer dans la base ou pour en sortir.

$$\begin{array}{rcl}
 x_5 & = & -0,5x_1 + 5,5x_2 + 2,5x_3 - 9x_4 \\
 x_6 & = & -0,5x_1 + 1,5x_2 + 0,5x_3 - x_4 \\
 x_7 & = & 1 - x_1 \\
 \hline
 z & = & 10x_1 - 57x_2 - 9x_3 - 24x_4
 \end{array}$$

Q1. En cas de choix pour une variable entrante, on prend la variable candidate pourvue du plus grand coefficient dans z (premier critère de Dantzig) et, en cas de choix pour une variable sortante, on prend la variable candidate de plus petit indice. Qu’observe-t-on ?

Q2. On applique la règle de Bland : en cas de choix pour une variable entrante ou sortante, on prend la variable candidate de plus petit indice.

5. Cet exemple est issu du livre de V. Chvátal, *op. cit.* On peut montrer que l’on doit avoir $n \geq 3$ et $m \geq 3$ pour qu’il puisse y avoir cyclage.

5.7. EXERCICES

65

Qu’observe-t-on ?

Corrigé.

Q1. On part du dictionnaire donné.

On fait entrer x_1 et sortir x_5 . Après la première itération :

$$\begin{array}{rclclclcl} x_1 & = & & 11x_2 & + & 5x_3 & - & 18x_4 & - & 2x_5 \\ x_6 & = & & -4x_2 & - & 2x_3 & + & 8x_4 & + & x_5 \\ x_7 & = & 1 & -11x_2 & - & 5x_3 & + & 18x_4 & + & 2x_5 \\ \hline z & = & & 53x_2 & + & 41x_3 & - & 204x_4 & - & 20x_5 \end{array}$$

On fait entrer x_2 et sortir x_6 . Après la deuxième itération :

$$\begin{array}{rclclclclcl} x_2 & = & & -0,5x_3 & + & 2x_4 & + & 0,25x_5 & - & 0,25x_6 \\ x_1 & = & & -0,5x_3 & + & 4x_4 & + & 0,75x_5 & - & 2,75x_6 \\ x_7 & = & 1 & +0,5x_3 & - & 4x_4 & - & 0,75x_5 & - & 2,75x_6 \\ \hline z & = & & 14,5x_3 & - & 98x_4 & - & 6,75x_5 & - & 13,25x_6 \end{array}$$

On fait entrer x_3 et sortir x_1 . Après la troisième itération :

$$\begin{array}{rclclclclcl} x_3 & = & & -2x_1 & + & 8x_4 & + & 1,5x_5 & - & 5,5x_6 \\ x_2 & = & & x_1 & - & 2x_4 & - & 0,5x_5 & + & 2,5x_6 \\ x_7 & = & 1 & -x_1 & & & & & & \\ \hline z & = & & -29x_1 & + & 18x_4 & + & 15x_5 & - & 93x_6 \end{array}$$

On fait entrer x_4 et sortir x_2 . Après la quatrième itération :

$$\begin{array}{rclclclclcl} x_4 & = & & 0,5x_1 & - & 0,5x_2 & - & 0,25x_5 & + & 1,25x_6 \\ x_3 & = & & 2x_1 & - & 4x_2 & - & 0,5x_5 & + & 4,5x_6 \\ x_7 & = & 1 & -x_1 & & & & & & \\ \hline z & = & & -20x_1 & - & 9x_2 & + & 10,5x_5 & - & 70,5x_6 \end{array}$$

On fait entrer x_5 et sortir x_3 . Après la cinquième itération :

$$\begin{array}{rclclclclcl} x_5 & = & & 4x_1 & - & 8x_2 & - & 2x_3 & + & 9x_6 \\ x_4 & = & & -0,5x_1 & + & 1,5x_2 & + & 0,5x_3 & - & x_6 \\ x_7 & = & 1 & -x_1 & & & & & & \\ \hline z & = & & 22x_1 & - & 93x_2 & - & 21x_3 & + & 24x_6 \end{array}$$

On fait entrer x_6 et sortir x_4 . Après la sixième itération :

$$\begin{array}{rcllclclcl}
 x_5 & = & & - & 0,5x_1 & + & 5,5x_2 & + & 2,5x_3 & - & 9x_4 \\
 x_6 & = & & - & 0,5x_1 & + & 1,5x_2 & + & 0,5x_3 & - & x_4 \\
 x_7 & = & 1 & - & x_1 & & & & & & \\
 \hline
 z & = & & & 10x_1 & - & 57x_2 & - & 9x_3 & - & 24x_4
 \end{array}$$

On retrouve le dictionnaire de départ : on observe qu'il y a cyclage. On peut remarquer que l'application du second critère de Dantzig au lieu du premier pour le choix des variables entrantes n'évite pas non plus le cyclage, puisque les étapes qu'on vient d'effectuer sont compatibles avec ce critère.

Q2. La règle de Bland donne les mêmes cinq premières itérations, mais pas la sixième. On reprend les calculs précédents après la cinquième itération :

$$\begin{array}{rcllclclcl}
 x_5 & = & & & 4x_1 & - & 8x_2 & - & 2x_3 & + & 9x_6 \\
 x_4 & = & & - & 0,5x_1 & + & 1,5x_2 & + & 0,5x_3 & - & x_6 \\
 x_7 & = & 1 & - & x_1 & & & & & & \\
 \hline
 z & = & & & 22x_1 & - & 93x_2 & - & 21x_3 & + & 24x_6
 \end{array}$$

On fait entrer x_1 (et non plus x_6) et sortir x_4 . Après la sixième itération :

$$\begin{array}{rcllclclcl}
 x_1 & = & & & 3x_2 & + & x_3 & - & 2x_4 & - & 2x_6 \\
 x_5 & = & & & 4x_2 & + & 2x_3 & - & 8x_4 & + & x_6 \\
 x_7 & = & 1 & - & 3x_2 & - & x_3 & + & 2x_4 & + & 2x_6 \\
 \hline
 z & = & & - & 27x_2 & + & x_3 & - & 44x_4 & - & 20x_6
 \end{array}$$

On fait entrer x_3 et sortir x_7 . Après la septième itération :

$$\begin{array}{rcllclclcl}
 x_3 & = & 1 & - & 3x_2 & + & 2x_4 & + & 2x_6 & - & x_7 \\
 x_1 & = & 1 & & & & & & & & - & x_7 \\
 x_5 & = & 2 & - & 2x_2 & - & 4x_4 & + & 5x_6 & - & 2x_7 \\
 \hline
 z & = & 1 & - & 30x_2 & - & 42x_4 & - & 18x_6 & - & 2x_7
 \end{array}$$

Tous les coefficients dans z sont négatifs ou nuls, la méthode s'arrête. On constate que l'application de la règle de Bland a permis d'éviter le cyclage.

5.7. EXERCICES

67

Exercice 4

Énoncé.

Q1. On considère le problème ci-dessous.

$$\begin{array}{l} \text{Maximiser } z = 5x_1 + 3x_2 \\ \text{avec les contraintes : } \left\{ \begin{array}{l} -4x_1 + 5x_2 \leq -10 \\ 5x_1 + 2x_2 \leq 10 \\ 3x_1 + 8x_2 \leq 12 \\ x_1 \geq 0, x_2 \geq 0. \end{array} \right. \end{array}$$

Montrer à l'aide de l'algorithme du simplexe que ce problème n'admet pas de solution réalisable.

Q2. On considère maintenant le problème ci-dessous (qui ne diffère du précédent que d'un signe dans la première contrainte). Le résoudre à l'aide de la méthode à deux phases issue de l'algorithme du simplexe.

$$\begin{array}{l} \text{Maximiser } z = 5x_1 + 3x_2 \\ \text{avec les contraintes : } \left\{ \begin{array}{l} -4x_1 - 5x_2 \leq -10 \\ 5x_1 + 2x_2 \leq 10 \\ 3x_1 + 8x_2 \leq 12 \\ x_1 \geq 0, x_2 \geq 0. \end{array} \right. \end{array}$$

Corrigé.

Les deux problèmes d'optimisation de cet exercice sont mis sous forme standard. On s'aperçoit que, dans les deux cas, la solution obtenue en mettant à zéro les deux variables x_1 et x_2 n'est pas réalisable. On utilise l'algorithme du simplexe à deux phases. La première phase débute par l'écriture du problème auxiliaire. Pour cela, on peut ajouter une variable x_0 dans les trois seconds membres des inégalités, comme pour l'exemple de la partie 5.6, page 58 ; on peut aussi se contenter d'ajouter cette variable x_0 aux seconds membres de valeurs négatives. C'est cette variante que nous choisissons ici afin de l'illustrer.

Q1. Le problème auxiliaire s'écrit alors, sous forme standard :

$$\begin{array}{l} \text{Maximiser } w = -x_0 \\ \text{avec les contraintes : } \left\{ \begin{array}{l} -4x_1 + 5x_2 - x_0 \leq -10 \\ 5x_1 + 2x_2 \leq 10 \\ 3x_1 + 8x_2 \leq 12 \\ x_0 \geq 0, x_1 \geq 0, x_2 \geq 0. \end{array} \right. \end{array}$$

On en déduit le dictionnaire initial :

$$\begin{array}{rclclcl} x_3 & = & -10 & + & x_0 & + & 4x_1 & - & 5x_2 \\ x_4 & = & 10 & & & - & 5x_1 & - & 2x_2 \\ x_5 & = & 12 & & & - & 3x_1 & - & 8x_2 \\ \hline w & = & & & - & x_0 & & & \end{array}$$

Ce dictionnaire n'est pas réalisable, mais on passe immédiatement à un dictionnaire réalisable en faisant entrer la variable x_0 et en faisant sortir la variable x_3 . On obtient le dictionnaire ci-dessous :

$$\begin{array}{rclclcl} x_0 & = & 10 & - & 4x_1 & + & 5x_2 & + & x_3 \\ x_4 & = & 10 & - & 5x_1 & - & 2x_2 & & \\ x_5 & = & 12 & - & 3x_1 & - & 8x_2 & & \\ \hline w & = & -10 & + & 4x_1 & - & 5x_2 & - & x_3 \end{array}$$

On fait maintenant entrer la variable x_1 et sortir la variable x_4 ; on obtient :

$$\begin{array}{rclclcl} x_1 & = & 2 & - & 2/5 x_2 & & - & 1/5 x_4 \\ x_0 & = & 2 & + & 33/5 x_2 & + & x_3 & + & 4/5 x_4 \\ x_5 & = & 6 & - & 34/5 x_2 & & + & 3/5 x_4 \\ \hline w & = & -2 & - & 33/5 x_2 & - & x_3 & - & 4/5 x_4 \end{array}$$

Il n'y a plus de variable entrante ; le maximum de w vaut -2 et n'est donc pas nul : le problème étudié n'admet pas de solution réalisable.

Q2. De la même façon que pour la question précédente, le problème auxiliaire s'écrit :

$$\begin{array}{l} \text{Maximiser } w = -x_0 \\ \text{avec les contraintes : } \left\{ \begin{array}{l} -4x_1 - 5x_2 - x_0 \leq -10 \\ 5x_1 + 2x_2 \leq 10 \\ 3x_1 + 8x_2 \leq 12 \\ x_1 \geq 0, x_2 \geq 0, x_0 \geq 0. \end{array} \right. \end{array}$$

On obtient le dictionnaire initial suivant :

$$\begin{array}{rclclcl} x_3 & = & -10 & + & x_0 & + & 4x_1 & + & 5x_2 \\ x_4 & = & 10 & & & - & 5x_1 & - & 2x_2 \\ x_5 & = & 12 & & & - & 3x_1 & - & 8x_2 \\ \hline w & = & & & - & x_0 & & & \end{array}$$

5.7. EXERCICES

69

Ce dictionnaire n'est pas réalisable mais, ici encore, on passe immédiatement à un dictionnaire réalisable en faisant entrer la variable x_0 et en faisant sortir la variable x_3 . On obtient le dictionnaire ci-dessous :

$$\begin{array}{rcll} x_0 & = & 10 & - 4x_1 - 5x_2 + x_3 \\ x_4 & = & 10 & - 5x_1 - 2x_2 \\ x_5 & = & 12 & - 3x_1 - 8x_2 \\ \hline w & = & -10 & + 4x_1 + 5x_2 - x_3 \end{array}$$

On fait maintenant entrer la variable x_1 et sortir la variable x_4 ; on obtient :

$$\begin{array}{rcll} x_1 & = & 2 & - 2/5 x_2 - 1/5 x_4 \\ x_0 & = & 2 & - 17/5 x_2 + x_3 + 4/5 x_4 \\ x_5 & = & 6 & - 34/5 x_2 + 3/5 x_4 \\ \hline w & = & -2 & + 17/5 x_2 - x_3 - 4/5 x_4 \end{array}$$

La variable x_2 est ici entrante alors que la variable x_0 sort. Le dictionnaire obtenu est :

$$\begin{array}{rcll} x_2 & = & 10/17 & + 5/17 x_3 + 4/17 x_4 - 5/17 x_0 \\ x_1 & = & 30/17 & - 2/17 x_3 - 5/17 x_4 + 2/17 x_0 \\ x_5 & = & 2 & - 2x_3 - x_4 + 2x_0 \\ \hline w & = & & - x_0 \end{array}$$

Le maximum du problème auxiliaire vaut 0 : le problème initial est réalisable. On peut maintenant commencer la seconde phase de la méthode. Pour obtenir un dictionnaire réalisable du problème initial, on reprend le dernier dictionnaire ci-dessus, duquel on supprime la variable x_0 et dans lequel on remplace la fonction w par la fonction z exprimée à l'aide des variables hors-base, c'est-à-dire de x_3 et x_4 . On obtient le dictionnaire ci-dessous :

$$\begin{array}{rcll} x_2 & = & 10/17 & + 5/17 x_3 + 4/17 x_4 \\ x_1 & = & 30/17 & - 2/17 x_3 - 5/17 x_4 \\ x_5 & = & 2 & - 2x_3 - x_4 \\ \hline z & = & 180/17 & + 5/17 x_3 - 13/17 x_4 \end{array}$$

La variable x_3 entre en base alors que la variable x_5 en sort. Le dictionnaire devient :

$$\begin{array}{rcll} x_3 & = & 1 & - 1/2 x_4 - 1/2 x_5 \\ x_2 & = & 15/17 & + 3/34 x_4 - 5/34 x_5 \\ x_1 & = & 28/17 & - 4/17 x_4 + 1/17 x_5 \\ \hline z & = & 185/17 & - 31/34 x_4 - 5/34 x_5 \end{array}$$

Ce dernier dictionnaire est optimal ; la solution optimale est donc donnée par :

- $x_1^* = 28/17$, $x_2^* = 15/17$ pour les variables de décision ;
- $x_3^* = 1$, $x_4^* = x_5^* = 0$ pour les variables d'écart ;
- $z^* = 185/17$ pour la fonction objectif.

Exercice 5

Énoncé. On considère un problème d'optimisation linéaire à une seule contrainte, défini par :

$$\text{Maximiser } \sum_{j=1}^n u_j x_j \text{ avec } \sum_{j=1}^n p_j x_j \leq P \text{ et } x_j \geq 0 \text{ pour } 1 \leq j \leq n.$$

Tous les coefficients u_j et p_j ainsi que P sont supposés strictement positifs. Montrer que la variable correspondant au plus grand rapport u_j/p_j est entrante et que, en la faisant entrer en base, on atteint le maximum de la fonction objectif en une seule itération. Exprimer ce maximum en fonction des différents coefficients.

Corrigé. Quitte à renuméroter les variables, on peut supposer que la variable x_1 correspond au plus grand rapport u_j/p_j : $j > 1 \Rightarrow u_j/p_j \leq u_1/p_1$. Le coefficient u_1 étant, par hypothèse, positif, la variable x_1 est entrante et on l'échange donc avec l'unique variable en base, qui correspond à l'unique contrainte, x_{n+1} . On avait :

$$x_{n+1} = P - \sum_{j=1}^n p_j x_j$$

et, après l'échange, on obtient :

$$x_1 = \frac{1}{p_1} \left(P - \sum_{j=2}^n p_j x_j - x_{n+1} \right).$$

En reportant cette valeur dans la fonction objectif, il vient :

$$z = \sum_{j=1}^n u_j x_j = \frac{u_1}{p_1} \left(P - \sum_{j=2}^n p_j x_j - x_{n+1} \right) + \sum_{j=2}^n u_j x_j$$

5.7. EXERCICES

71

ou encore :

$$z = \frac{u_1 P}{p_1} + \sum_{j=2}^n \left(u_j - \frac{u_1 p_j}{p_1} \right) x_j - \frac{u_1}{p_1} x_{n+1}.$$

Compte tenu de la numérotation adoptée, les coefficients de toutes les variables qui interviennent dans l’écriture de z sont négatifs ou nuls. On a donc déterminé la valeur maximum de z en une itération, et cette valeur maximum est égale à $\frac{u_1 P}{v_1}$: cela revient à saturer la contrainte avec la variable pour laquelle le rapport u_j/p_j est maximum.

Chapitre 6

Dualité en optimisation linéaire

6.1 Définition du problème dual

Remarque

Nous ne considérons dans ce chapitre que les **problèmes d’optimisation linéaire écrits sous forme standard**. Pour définir le problème dual d’un problème quelconque d’optimisation linéaire, on peut le mettre sous forme standard avant de déterminer le problème dual, comme indiqué dans le chapitre 5. Un exemple est donné en exercice.

On considère donc le problème (P) :

$$\begin{aligned} &\text{Maximiser } z = \sum_{j=1}^n c_j x_j \\ &\text{avec les contraintes : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n a_{ij} x_j \leq b_i \\ \text{pour } j \in \{1, 2, \dots, n\}, x_j \geq 0. \end{cases} \end{aligned}$$

S’il existe m réels y_i positifs ou nuls tels que, pour tout $j \in \{1, 2, \dots, n\}$, $\sum_{i=1}^m a_{ij} y_i \geq c_j$, alors on a, pour toute solution réalisable (x_1, \dots, x_n) de (P) :

$$\sum_{j=1}^n c_j x_j \leq \sum_{j=1}^n \left(\sum_{i=1}^m a_{ij} y_i \right) x_j = \sum_{i=1}^m \left(\sum_{j=1}^n a_{ij} x_j \right) y_i \leq \sum_{i=1}^m b_i y_i.$$

D’où :

$$\sum_{j=1}^n c_j x_j \leq \sum_{i=1}^m b_i y_i$$

et cette dernière quantité donne donc un majorant de la fonction objectif. Le *problème dual* ((D)) du problème ((P)) s'écrit :

$$\begin{aligned} & \text{Minimiser } \sum_{i=1}^m b_i y_i \\ & \text{avec les contraintes : } \begin{cases} \text{pour } j \in \{1, 2, \dots, n\}, \sum_{i=1}^m a_{ij} y_i \geq c_j \\ \text{pour } i \in \{1, 2, \dots, m\}, y_i \geq 0. \end{cases} \end{aligned}$$

Le problème ((P)) prend alors le nom de *problème primal*. On voit que, pour toute solution réalisable y_1^*, \dots, y_m^* du dual (c'est-à-dire satisfaisant les contraintes de ((D))), $\sum_{i=1}^m b_i y_i^*$ est un majorant de la fonction objectif de ((P)).

Remarque

On établit facilement que le problème dual de ((D)) est ((P)).

6.2 Théorème de la dualité

De la définition du problème dual, nous déduisons immédiatement la proposition suivante :

Proposition 6.1. Soient $(x_1^*, x_2^*, \dots, x_n^*)$ une solution réalisable du problème primal et $(y_1^*, y_2^*, \dots, y_m^*)$ une solution réalisable du problème dual. On a :

$$\sum_{j=1}^n c_j x_j^* \leq \sum_{i=1}^m b_i y_i^*$$

De plus, si les deux quantités ci-dessus sont égales, alors $x_1^*, x_2^*, \dots, x_n^*$ constituent une solution optimale du problème primal et $y_1^*, y_2^*, \dots, y_m^*$ une solution optimale du problème dual.

Application

La considération du problème dual nous permet de vérifier que nous avons bien trouvé, par l'algorithme du simplexe, une solution optimale pour un problème donné. Nous allons l'expliquer sur le problème traité dans le chapitre 5.

6.2. THÉORÈME DE LA DUALITÉ

75

Le problème (P) est :

$$\begin{aligned} & \text{Maximiser } z = 7x_1 + 9x_2 + 18x_3 + 17x_4 \\ & \text{avec les contraintes : } \begin{cases} 2x_1 + 4x_2 + 5x_3 + 7x_4 \leq 42 \\ x_1 + x_2 + 2x_3 + 2x_4 \leq 17 \\ x_1 + 2x_2 + 3x_3 + 3x_4 \leq 24 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0. \end{cases} \end{aligned}$$

Nous avons établi que l'optimum de ce problème vaut $z^* = 147$ et est obtenu pour $x_1^* = 3, x_2^* = 0, x_3^* = 7, x_4^* = 0$. Nous voulons ici vérifier ce résultat.

Le problème dual (D) s'écrit :

$$\begin{aligned} & \text{Minimiser } 42y_1 + 17y_2 + 24y_3 \\ & \text{avec les contraintes : } \begin{cases} 2y_1 + y_2 + y_3 \geq 7 \\ 4y_1 + y_2 + 2y_3 \geq 9 \\ 5y_1 + 2y_2 + 3y_3 \geq 18 \\ 7y_1 + 2y_2 + 3y_3 \geq 17 \\ y_1 \geq 0, y_2 \geq 0, y_3 \geq 0. \end{cases} \end{aligned}$$

Rappelons que, dans le dernier dictionnaire, la fonction objectif s'écrivait :

$$z = 147 - 2x_2 - x_4 - 3x_6 - 4x_7.$$

Considérons les valeurs $y_1^* = 0, y_2^* = 3, y_3^* = 4$. Ces valeurs ne sont pas choisies au hasard : ce sont les opposés des coefficients respectivement de x_5, x_6, x_7 dans l'expression ci-dessus de z ; nous justifierons ce choix plus loin.

On a : $42y_1^* + 17y_2^* + 24y_3^* = 147$.

Par ailleurs, on vérifie aisément que les y_i^* satisfont les contraintes du problème dual, donc constituent une solution réalisable du dual.

La proposition ci-dessus nous permet d'affirmer que la valeur 147 est l'optimum du problème primal : ayant trouvé une solution réalisable du dual qui donne à la fonction objectif du dual la valeur que la solution trouvée pour le primal donnait à la fonction objectif du primal, nous pouvons affirmer que nous avons trouvé le maximum de la fonction objectif du primal et que nous avons également trouvé le minimum de la fonction objectif du dual. Cette vérification constitue donc un *certificat d'optimalité* de la solution trouvée pour le primal.

Cette proposition a pour corollaire ce qui suit :

Proposition 6.2. Si le problème primal admet une solution réalisable et est non borné, le problème dual n'admet pas de solution réalisable.

Preuve. Supposons que le problème dual admette une solution réalisable et notons w^* la valeur correspondante de la fonction objectif du problème dual. La fonction objectif du problème primal est alors majorée par w^* , ce qui contredit l’hypothèse. \diamond

Remarques

1. En appliquant ce qui précède au problème dual, on a aussi le résultat suivant : si le problème dual admet une solution réalisable et est non borné, le problème primal n’admet pas de solution réalisable.
2. Par contraposée, on obtient l’implication suivante : si (D) (respectivement (P)) admet une solution réalisable, alors (P) (respectivement (D)) n’en admet pas – et dans ce cas (D) (respectivement (P)) n’est pas borné – ou est borné.
3. Il résulte de ce qui précède que (P) et (D) ne peuvent pas être simultanément non bornés.
4. Il existe des cas pour lesquels (P) et (D) sont simultanément non réalisables.

Le théorème suivant, parfois appelé *théorème fondamental de la dualité*, généralise les constatations de l’application faite ci-dessus.

Théorème 6.3. (de la dualité) Si le problème primal a une solution optimale $x_1^*, x_2^*, \dots, x_n^*$, alors le problème dual a une solution optimale $y_1^*, y_2^*, \dots, y_m^*$ et $\sum_{j=1}^n c_j x_j^* = \sum_{i=1}^m b_i y_i^*$ (autrement dit, le maximum primal est égal au minimum dual).

Nous allons prouver ce théorème fondamental en même temps que la proposition suivante :

Proposition 6.4. Si le problème primal admet une solution optimale et si l’expression de la fonction objectif du primal dans le dernier dictionnaire obtenu par la méthode du simplexe s’écrit :

$$z = z^* + \sum_{k=1}^{n+m} d_k x_k$$

(où x_{n+i} représente la i^{e} variable d’écart), alors une solution optimale du problème dual est donnée par $y_i^* = -d_{n+i}$.

6.2. THÉORÈME DE LA DUALITÉ

77

Preuve du théorème de la dualité et de la proposition

Supposons le primal résolu par la méthode du simplexe, exposée dans le chapitre I. Aux n variables initiales du problème nous avons ajouté m variables d'écart x_{n+1}, \dots, x_{n+m} . À la i^{e} contrainte du primal sont associées la variable d'écart x_{n+i} et la variable y_i du dual, ce qui établit un lien canonique entre x_{n+i} et y_i . Considérons l'expression de la fonction objectif du primal dans le dernier dictionnaire du simplexe primal :

$$z = z^* + \sum_{k=1}^{n+m} d_k x_k.$$

Les d_k sont tous négatifs ou nuls (puisqu'il s'agit du dernier dictionnaire) et les d_k associés aux variables en base sont nuls.

Par ailleurs, on a $z^* = \sum_{j=1}^n c_j x_j^*$ par définition de z et $x_{n+i} = b_i - \sum_{j=1}^n a_{ij} x_j$ par définition des variables d'écart.

Posons, pour $i \in \{1, \dots, m\}$, $y_i^* = -d_{n+i}$; on a alors : $y_i^* \geq 0$. On a de plus, en distinguant dans z les variables d'écart des autres :

$$z = z^* + \sum_{j=1}^n d_j x_j - \sum_{i=1}^m \left(b_i - \sum_{j=1}^n a_{ij} x_j \right) y_i^*,$$

ou encore :

$$z = z^* - \sum_{i=1}^m b_i y_i^* + \sum_{j=1}^n \left(d_j + \sum_{i=1}^m a_{ij} y_i^* \right) x_j.$$

Mais, par définition de z , on a aussi : $z = \sum_{j=1}^n c_j x_j$.

À cause de l'indépendance des variables x_j , on déduit de ces égalités :

$$\begin{cases} z^* = \sum_{i=1}^m b_i y_i^* \\ \text{pour } j \in \{1, \dots, n\}, c_j = d_j + \sum_{i=1}^m a_{ij} y_i^* \end{cases}$$

Les d_j ($j \in \{1, \dots, n+m\}$) étant négatifs ou nuls, on obtient finalement :

$$\begin{cases} \text{pour } j \in \{1, \dots, n\}, \sum_{i=1}^m a_{ij} y_i^* \geq c_j \\ \text{pour } i \in \{1, \dots, m\}, y_i^* \geq 0. \end{cases}$$

Les nombres $y_1^*, y_2^*, \dots, y_m^*$ forment donc une solution réalisable du problème dual qui donne à la fonction objectif du problème dual la valeur z^* . La proposition du début de ce paragraphe permet de conclure. \diamond

6.3 Le théorème des écarts complémentaires : un certificat d’optimalité

L’application exposée dans le paragraphe précédent donne une méthode pour démontrer l’optimalité d’une solution du problème primal mais nécessite la connaissance du dernier dictionnaire de la méthode du simplexe. Nous allons voir que l’on peut aussi réussir à fournir un *certificat d’optimalité* du primal, en connaissant seulement les valeurs x_1^*, \dots, x_n^* qui donnent son maximum à l’objectif du primal.

Théorème 6.5. (des écarts complémentaires) Une solution réalisable x_1^*, \dots, x_n^* du primal est optimale si et seulement s’il existe des nombres y_1^*, \dots, y_m^* vérifiant ce qui suit :

- pour $i \in \{1, \dots, m\}$, si $\sum_{j=1}^n a_{ij}x_j^* < b_i$, alors $y_i^* = 0$
- pour $j \in \{1, \dots, n\}$, si $x_j^* > 0$, alors $\sum_{i=1}^m a_{ij}y_i^* = c_j$

et constituant une solution réalisable du problème dual :

$$\begin{cases} \text{pour } j \in \{1, \dots, n\}, \sum_{i=1}^m a_{ij}y_i^* \geq c_j \\ \text{pour } i \in \{1, \dots, m\}, y_i^* \geq 0. \end{cases}$$

De plus, ces nombres y_1^*, \dots, y_m^* constituent une solution optimale du dual.

Avant de donner la preuve de ce théorème nous allons l’appliquer à l’exemple du chapitre I. Considérons la déclaration :

« $x_1^* = 3, x_2^* = 0, x_3^* = 7, x_4^* = 0$ constituent une solution optimale du primal ».

On vérifie aisément que ces valeurs définissent bien une solution réalisable

6.3. LE THÉORÈME DES ÉCARTS COMPLÉMENTAIRES : UN CERTIFICAT D’OPTIMALITÉ

du problème primal. Cherchons donc s’il existe y_1^*, y_2^*, y_3^* vérifiant :

$$\begin{cases} y_1^* = 0 & \text{puisque la première contrainte du problème « n'est pas saturée »} \\ 2y_1^* + y_2^* + y_3^* = 7 & \text{puisque } x_1^* > 0 \\ 5y_1^* + 2y_2^* + 3y_3^* = 18 & \text{puisque } x_3^* > 0. \end{cases}$$

Utilisant la nullité de y_1^* , on obtient :

$$\begin{cases} y_2^* + y_3^* = 7 \\ 2y_2^* + 3y_3^* = 18. \end{cases}$$

La résolution de ce système donne $y_2^* = 3, y_3^* = 4$. Ces valeurs satisfont bien les contraintes du problème dual. En effet :

$$\begin{aligned} 4y_1^* + y_2^* + 2y_3^* &= 11 \geq 9 \\ \text{et } 7y_1^* + 2y_2^* + 3y_3^* &= 18 \geq 17. \end{aligned}$$

Les deux autres inégalités du même type résultent du système définissant y_1^*, y_2^* et y_3^* . Enfin y_1^*, y_2^*, y_3^* sont positifs ou nuls.

La solution proposée pour le primal est donc bien optimale. On vérifie d’autre part que y_1^*, y_2^*, y_3^* donnent le maximum primal à la fonction objectif duale : ces y_1^*, y_2^*, y_3^* constituent donc bien une solution optimale du problème dual.

Preuve du théorème des écarts complémentaires.

La preuve se déduit immédiatement du résultat que nous énonçons puis prouvons ci-dessous.

Si on connaît une solution réalisable (x_j^*) du primal et une solution réalisable (y_i^*) du dual, ces solutions sont optimales si et seulement si :

- pour $j \in \{1, \dots, n\}$, $x_j^* = 0$ ou $\sum_{i=1}^m a_{ij}y_i^* = c_j$
- et, pour $i \in \{1, \dots, m\}$, $y_i^* = 0$ ou $\sum_{j=1}^n a_{ij}x_j^* = b_i$.

En effet, d’après le théorème de la dualité, on sait que si on connaît une solution réalisable (x_j^*) du primal et une solution réalisable (y_i^*) du dual, ces solutions sont optimales si et seulement si on a :

$$\sum_{j=1}^n c_j x_j^* = \sum_{i=1}^m b_i y_i^*.$$

Or, on a les inégalités suivantes (conséquences de la réalisabilité des solutions) :

$$\sum_{j=1}^n c_j x_j^* \leq \sum_{j=1}^n \left(\sum_{i=1}^m a_{ij} y_i^* \right) x_j^* = \sum_{i=1}^m \left(\sum_{j=1}^n a_{ij} x_j^* \right) y_i \leq \sum_{i=1}^m b_i y_i^*.$$

Si, avant sommation, une des inégalités était stricte, il en serait de même après sommation. On voit donc qu'il y a égalité entre les bornes de cette suite si et seulement si on a :

- pour $j \in \{1, \dots, n\}$, $x_j^* = 0$ ou $\sum_{i=1}^m a_{ij} y_i^* = c_j$
- et, pour $i \in \{1, \dots, m\}$, $y_i^* = 0$ ou $\sum_{j=1}^n a_{ij} x_j^* = b_i$. ◇

On peut remarquer que, si l'on peut déterminer les y_i de façon unique, dès lors que l'une des inégalités requises (y compris les contraintes de signe) n'est pas vérifiée, on peut en déduire que la solution n'est pas optimale.

6.4 La signification économique du dual

Nous allons montrer ici que la connaissance de la solution du problème dual peut permettre de prendre en compte des données économiques.

Nous allons considérer que :

- b_i représente la quantité totale de la ressource i ;
- a_{ij} représente la quantité de la ressource i consommée par la fabrication d'une unité de produit j ;
- x_j représente la quantité fabriquée de produit j ;
- c_j représente la valeur unitaire du produit j .

La relation à l'optimum : $z^* = \sum_{j=1}^n c_j x_j^* = \sum_{i=1}^m b_i y_i^*$ induit que y_i doit représenter la « valeur unitaire de la ressource i ». Ces variables duales y_i sont souvent appelées *prix implicite*. La valeur de y_i donne le montant maximum que l'on serait prêt à payer pour obtenir une unité supplémentaire de la ressource i .

6.4. LA SIGNIFICATION ÉCONOMIQUE DU DUAL

81

Les relations $\sum_{i=1}^m a_{ij}y_i \geq c_j$, ($j \in \{1, \dots, n\}$) peuvent se comprendre à l'aide du schéma suivant : supposons qu'une personne étrangère à l'entreprise souhaite acquérir les ressources de l'entreprise ; elle doit proposer pour les ressources un prix tel que ce soit plus intéressant pour l'entreprise de lui vendre ses ressources que de fabriquer elle-même les produits (or, c_j est le profit escompté sur le produit j) et bien sûr elle désire faire cet achat des ressources à un prix minimum. Rappelons que le coefficient a_{ij} représente la quantité de la ressource i requise pour fabriquer une unité de produit j de sorte que $\sum_{i=1}^m a_{ij}y_i$ représente la somme à dépenser pour acquérir les ressources nécessaires à la fabrication d'une unité du produit j .

Nous allons donner un second éclairage à l'aide de notre exemple.

Problème

Le fabricant de tissus du chapitre 1 a la possibilité de faire faire à ses ouvriers spécialisés dans la teinture quelques heures supplémentaires à un prix horaire de t euros. A-t-il ou non intérêt à utiliser cette possibilité ?

Pour résoudre ce problème, nous allons énoncer un théorème, que nous démontrerons après avoir résolu notre problème.

Théorème 6.6. On considère le problème (P) :

$$\begin{aligned} \text{Maximiser } z &= \sum_{j=1}^n c_j x_j \\ \text{avec les contraintes : } &\begin{cases} \text{pour } i \in \{1, \dots, m\}, \sum_{j=1}^n a_{ij} x_j \leq b_i \\ \text{pour } j \in \{1, 2, \dots, n\}, x_j \geq 0. \end{cases} \end{aligned}$$

On suppose que la base optimale de (P) est non dégénérée. Pour des variations δb_i des b_i , on considère le problème (P_δ) défini par :

$$\begin{aligned} \text{Maximiser } z &= \sum_{j=1}^n c_j x_j \\ \text{avec les contraintes : } &\begin{cases} \text{pour } i \in \{1, \dots, m\}, \sum_{j=1}^n a_{ij} x_j \leq b_i + \delta b_i \\ \text{pour } j \in \{1, 2, \dots, n\}, x_j \geq 0. \end{cases} \end{aligned}$$

On suppose que les variations δb_i sont suffisamment faibles pour que la base optimale de (P) soit encore réalisable pour (P_δ) . La variation de la valeur optimum de la fonction objectif du programme linéaire vaut alors $\sum_{i=1}^m \delta b_i y_i^*$ où (y_1^*, \dots, y_m^*) est solution optimale du problème dual de (P) .

Remarque. En approfondissement la preuve du théorème de la dualité, on obtiendrait que la non dégénérescence de la base optimale de (P) implique l'unicité de la solution du problème dual.

Pour notre problème, appelons u le nombre d'heures supplémentaires pour la teinture (avec u petit). La variation du second membre est $(0, 0, u)$. La solution optimale du problème dual est $(0, 3, 4)$. La variation de la fonction objectif est donc égale à $4u$. Il s'agit donc de la variation du chiffre d'affaires que le patron peut espérer de u heures supplémentaires, mais ce n'est pas là un bénéfice net puisqu'elles lui coûteront $t.u$ euros.

On voit qu'il a intérêt à recourir à cette solution dès que l'on a $t \leq 4$ €. On retrouve là l'interprétation de y_i^* : valeur unitaire de la ressource.

Preuve du théorème. On considère la suite des dictionnaires obtenus lorsqu'on résout (P) par la méthode du simplexe. Quand on change b pour $b + \delta b$, seules les constantes des seconds membres sont changées. Si le dernier dictionnaire reste réalisable (c'est-à-dire si les constantes des seconds membres des égalités exprimant les variables de base restent positives ou nulles), alors ce dernier dictionnaire reste optimal. On suppose qu'on est dans ce cas. Les coefficients des variables hors-bases dans la ligne exprimant la fonction z étant inchangés, la solution du problème dual est inchangée. La valeur optimale commune des nouveaux problèmes primal et dual vaut : $\sum_{i=1}^m (b_i + \delta b_i) y_i^* = \sum_{i=1}^m b_i y_i^* + \sum_{i=1}^m \delta b_i y_i^*$ où (y_1^*, \dots, y_m^*) est solution optimale du problème dual de P . La variation de la valeur optimum de la fonction objectif vaut $\sum_{i=1}^m \delta b_i y_i^*$. \diamond

On admet que si la base optimale de (P) est non dégénérée, pour des raisons de continuité, il existe des variations non nulles des δb_i assez petites pour conserver le fait que la base optimale de (P) reste réalisable.

6.5 Problème dual-réalisable

L'utilisation du problème dual permet, sans utiliser l'algorithme à deux phases décrit dans le premier chapitre, de résoudre un problème d'optimisation linéaire où la solution nulle n'est pas réalisable, pourvu que les coefficients c_j de la fonction objectif du problème écrit sous forme standard soient tous négatifs ou nuls. Un tel problème est dit dual-réalisable.

Exemple

Considérons le problème d'optimisation linéaire :

$$\begin{aligned} &\text{Minimiser } x_1 + x_2 \\ &\text{avec les contraintes : } \begin{cases} 3x_1 + x_2 \geq 4 \\ -7x_1 + x_2 \geq -7 \\ x_1 \geq 0, x_2 \geq 0 \end{cases} \end{aligned}$$

dont l'écriture, sous forme standard est :

$$\begin{aligned} &\text{Maximiser } -x_1 - x_2 \\ &\text{avec les contraintes : } \begin{cases} -3x_1 - x_2 \leq -4 \\ 7x_1 - x_2 \leq 7 \\ x_1 \geq 0, x_2 \geq 0. \end{cases} \end{aligned}$$

Le problème dual s'écrit :

$$\begin{aligned} &\text{Minimiser } -4y_1 + 7y_2 \\ &\text{avec les contraintes : } \begin{cases} -3y_1 + 7y_2 \geq -1 \\ -y_1 - y_2 \geq -1 \\ y_1 \geq 0, y_2 \geq 0 \end{cases} \end{aligned}$$

ou encore :

$$\begin{aligned} &\text{Maximiser } 4y_1 - 7y_2 \\ &\text{avec les contraintes : } \begin{cases} 3y_1 - 7y_2 \leq 1 \\ y_1 + y_2 \leq 1 \\ y_1 \geq 0, y_2 \geq 0. \end{cases} \end{aligned}$$

Les variables d'écarts constituent maintenant une base réalisable : la méthode du simplexe ne nécessite qu'une seule phase. De la solution du problème dual, on pourra déduire la solution du problème primal.

6.6 Exercices

Exercice 1

Énoncé. On considère le problème :

$$\begin{aligned} & \text{Maximiser } z = 4x_1 + 3x_2 \\ & \text{avec les contraintes : } \begin{cases} 5x_1 + 3x_2 \leq 30 \\ 2x_1 + 3x_2 \leq 24 \\ x_1 + 3x_2 \leq 18 \\ x_1 \geq 0, x_2 \geq 0. \end{cases} \end{aligned}$$

Q1. Résoudre graphiquement ce problème.

Q2. Utiliser le théorème des écarts complémentaires pour prouver que la solution graphique est exacte.

Q3. La fonction z donne un profit en euros. On envisage de se procurer une unité de plus de la première ressource à un prix unitaire de t euros. Jusqu'à quelle valeur de t cela semble-t-il intéressant ?

Q4. On suppose qu'on se procure a unités supplémentaires de la première ressource. Jusqu'à quelle valeur de a la base optimale du problème initial reste-t-elle réalisable (auquel cas cette base reste optimale) ?

Corrigé.

Q1. On représente graphiquement le problème par la figure 6.1. La solution graphique est : $x_1^* = 3, x_2^* = 5$.

Q2. On vérifie la solution graphique en utilisant le théorème des écarts complémentaires. La solution $x_1^* = 3, x_2^* = 5$ est bien une solution réalisable. On cherche y_1^*, y_2^* et y_3^* vérifiant les conditions du théorème des écarts complémentaires.

- Avec $x_1^* = 3$ et $x_2^* = 5$, on a : $2x_1^* + 3x_2^* = 21 < 24$, ce qui entraîne : $y_2^* = 0$.
- Comme x_1^* et x_2^* sont non nuls, on doit avoir :

$$\begin{cases} 5y_1^* + 2y_2^* + y_3^* = 4 \\ 3y_1^* + 2y_2^* + 3y_3^* = 3. \end{cases}$$

Avec $y_2^* = 0$, le système ci-dessus a pour unique solution : $y_1^* = 3/4, y_3^* = 1/4$. Il reste à vérifier que les valeurs $y_1^* = 3/4, y_2^* = 0, y_3^* = 1/4$ constituent une

6.6. EXERCICES

85

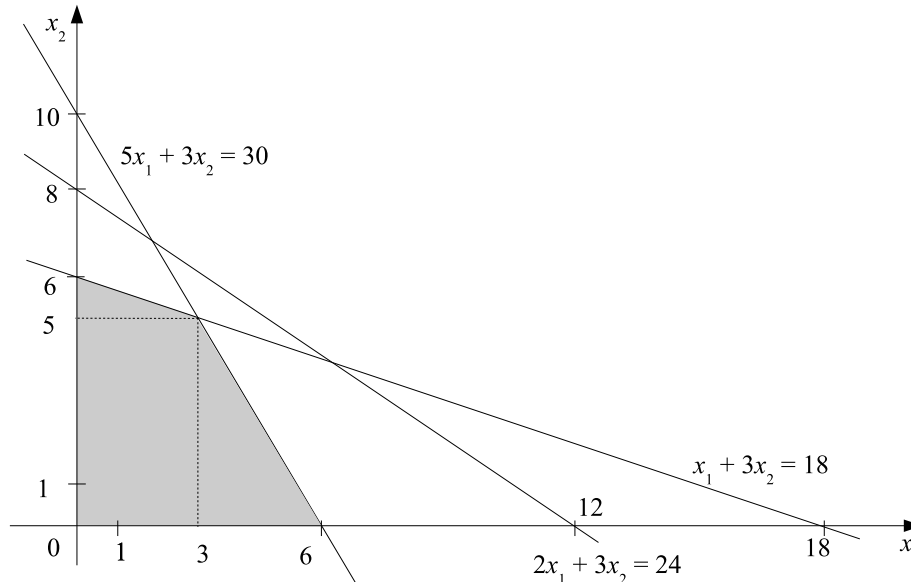


FIGURE 6.1 – Solution graphique.

solution réalisable du problème dual, ce qui est immédiat. La solution déterminée graphiquement est bien optimale.

Q3. La valeur marginale de la première ressource est égale à $3/4$. Se procurer une unité de plus de la première ressource est intéressant si le prix unitaire de celle-ci est inférieure à $0,75$ €.

Q4. On note x_3 , x_4 et x_5 les trois variables d'écart. On remarque que dans la solution basique optimale du problème initial, on a $x_3^* = 0$, $x_4^* = 3$, $x_5^* = 0$. En ajoutant a à la première ressource, on obtient :

$$\begin{cases} 5x_1 + 3x_2 + x_3 = 30 + a \\ 2x_1 + 3x_2 + x_4 = 24 \\ x_1 + 3x_2 + x_5 = 18. \end{cases}$$

En s'appuyant sur la solution numérique du problème initial, posons $x_1 = 3 + \delta x_1$, $x_2 = 5 + \delta x_2$, $x_4 = 3 + \delta x_4$. On a alors :

$$\begin{cases} 5\delta x_1 + 3\delta x_2 = a \\ 2\delta x_1 + 3\delta x_2 + \delta x_4 = 0 \\ \delta x_1 + 3\delta x_2 = 0. \end{cases}$$

Ce système a pour solution : $\delta x_1 = \frac{a}{4}$, $\delta x_2 = -\frac{a}{12}$, $\delta x_4 = -\frac{a}{4}$.

La solution est réalisable si :

$$\begin{cases} 3 + \frac{a}{4} \geq 0 \\ 5 - \frac{a}{12} \geq 0 \\ 3 - \frac{a}{4} \geq 0 \end{cases}$$

ce qui équivaut à : $a \leq 12$.

Exercice 2

Énoncé. On propose $x_1^* = 0$, $x_2^* = \frac{4}{3}$, $x_3^* = \frac{2}{3}$, $x_4^* = \frac{5}{3}$, $x_5^* = 0$ comme solution optimale du problème suivant :

$$\begin{aligned} & \text{Maximiser } z = 7x_1 + 6x_2 + 5x_3 - 2x_4 + 3x_5 \\ & \text{avec les contraintes : } \begin{cases} x_1 + 3x_2 + 5x_3 - 2x_4 + 2x_5 \leq 4 \\ 4x_1 + 2x_2 - 2x_3 + x_4 + x_5 \leq 3 \\ 2x_1 + 4x_2 + 4x_3 - 2x_4 + 5x_5 \leq 5 \\ 3x_1 + x_2 + 2x_3 - x_4 - 2x_5 \leq 1 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0. \end{cases} \end{aligned}$$

Est-ce correct ?

Corrigé.

La vérification se fait comme suit. On examine d’abord si la solution proposée est réalisable.

- La solution proposée est positive ou nulle.
- On vérifie qu’elle satisfait les autres contraintes et repérons simultanément les contraintes saturées et celles qui ne le sont pas.
 - $x_1^* + 3x_2^* + 5x_3^* - 2x_4^* + 2x_5^* = 4$: contrainte saturée.
 - $4x_1^* + 2x_2^* - 2x_3^* + x_4^* + x_5^* = 3$: contrainte saturée.
 - $2x_1^* + 4x_2^* + 4x_3^* - 2x_4^* + 5x_5^* = 14/3 < 5$: contrainte vérifiée mais non saturée.
 - $3x_1^* + x_2^* + 2x_3^* - x_4^* - 2x_5^* = 1$: contrainte saturée.
- On écrit les égalités que doivent vérifier les nombres y_i^* ($i = 1, 2, 3, 4$).

6.6. EXERCICES

87

- Puisque la troisième contrainte n'est pas saturée, $y_3^* = 0$.
- Puisque $x_2^* > 0$, $3y_1^* + 2y_2^* + 4y_3^* + y_4^* = 6$.
- Puisque $x_3^* > 0$, $5y_1^* - 2y_2^* + 4y_3^* + 2y_4^* = 5$.
- Puisque $x_4^* > 0$, $-2y_1^* + y_2^* - 2y_3^* - y_4^* = -2$.
- On calcule les y_i^* ($i = 1, 2, 4$) :

$$\begin{cases} 3y_1^* + 2y_2^* + y_4^* = 6 \\ 5y_1^* - 2y_2^* + 2y_4^* = 5 \\ -2y_1^* + y_2^* - y_4^* = -2 \end{cases}$$

La solution de ce système est : $y_1^* = y_2^* = y_4^* = 1$.

- On regarde si les y_i^* ($i = 1, 2, 3, 4$) constituent une solution réalisable du problème dual.
- Ils sont tous positifs ou nuls.
- Il reste à vérifier la première et la cinquième contrainte du problème dual puisque les autres contraintes sont saturées par définition des y^* :

$$\begin{aligned} y_1^* + 4y_2^* + 2y_3^* + 3y_4^* &= 8 \geq 7 \\ 2y_1^* + y_2^* + 5y_3^* - 2y_4^* &= 1 < 3 \end{aligned}$$

La dernière contrainte du dual n'est pas vérifiée : la solution actuelle n'est pas optimale.

On peut remarquer que, si on veut maintenant rechercher la solution optimale, il serait judicieux de partir de la base x_2, x_3, x_4, x_8 , où x_8 représente la troisième variable d'écart ; cette base correspond à la solution proposée.

Exercice 3

Énoncé. Donner un exemple de problème (P) tel que ni le problème (P) et le problème dual de (P) n'admettent de solution réalisable.

Corrigé. On considère le problème (P) suivant :

$$\begin{aligned} &\text{Maximiser } z = 2x_1 - x_2 \\ &\begin{cases} x_1 - x_2 \leq 1 \\ -x_1 + x_2 \leq -2 \\ x_1 \geq 0, x_2 \geq 0 \end{cases} \end{aligned}$$

Le dual (Q) de (P) est :

$$\begin{aligned} &\text{Minimiser } w = y_1 - 2y_2 \\ &\begin{cases} y_1 - y_2 \geq 2 \\ -y_1 + y_2 \geq -1 \\ y_1 \geq 0, y_2 \geq 0 \end{cases} \end{aligned}$$

On vérifie aisément que les problèmes (P) et (Q) n'admettent pas de solution réalisable.

Exercice 4

Énoncé.

Q1. On considère le problème (P) ci-dessous :

$$\begin{aligned} &\text{Minimiser } z = \sum_{j=1}^n c_j x_j \\ &\text{avec : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n a_{ij} x_j \geq b_i \\ \text{pour } i \in \{m+1, \dots, m+p\}, \sum_{j=1}^n a_{ij} x_j = b_i. \\ \text{pour } j \in \{1, \dots, n\}, x_j \in \mathbb{R}. \end{cases} \end{aligned}$$

Montrer que le problème (Q) défini ci-dessous est le problème dual de (P) :

$$\begin{aligned} &\text{Maximiser } w = \sum_{i=1}^{m+p} b_i y_i \\ &\text{avec : } \begin{cases} \text{pour } j \in \{1, 2, \dots, n\}, \sum_{i=1}^{m+p} a_{ij} y_i = c_j \\ \text{pour } i \in \{1, 2, \dots, m\}, y_i \geq 0 \\ \text{pour } i \in \{m+1, \dots, m+p\}, y_i \in \mathbb{R}. \end{cases} \end{aligned}$$

Q2. Établir le théorème suivant (théorème de Farkas)¹ stipulant que les deux propositions ci-dessous sont équivalentes.

1. Ce théorème sera utilisé dans le chapitre 9 pour établir les conditions de Karush, Kuhn et Tucker.

6.6. EXERCICES

89

$$(i) \text{ Soit } x \in \mathbb{R}^n; \text{ si on a : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n a_{ij}x_j \geq 0 \\ \text{pour } i \in \{m+1, \dots, m+p\}, \sum_{j=1}^n a_{ij}x_j = 0 \end{cases}$$

alors : $\sum_{j=1}^n c_j x_j \geq 0$.

$$(ii) \text{ Il existe } y \in \mathbb{R}^{m+p} \text{ vérifiant : } \begin{cases} \text{pour } j \in \{1, 2, \dots, n\}, \sum_{i=1}^{m+p} a_{ij}y_i = c_j \\ \text{pour } i \in \{1, 2, \dots, m\}, y_i \geq 0. \end{cases}$$

Corrigé.

Q1. On commence par se rapprocher de la forme standard :

$$\begin{aligned} & \text{Maximiser } \sum_{j=1}^n (-c_j x_j) \\ & \text{avec : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n (-a_{ij}x_j) \leq -b_i \\ \text{pour } i \in \{m+1, \dots, m+p\}, \sum_{j=1}^n (-a_{ij}x_j) \leq -b_i \\ \text{pour } i \in \{m+1, \dots, m+p\}, \sum_{j=1}^n a_{ij}x_j \leq b_i \\ \text{pour } j \in \{1, 2, \dots, n\}, x_j \in \mathbb{R}. \end{cases} \end{aligned}$$

Mettons maintenant le problème sous forme standard. Pour $j \in \{1, 2, \dots, n\}$, on pose : $x_j = x_j^1 - x_j^2$ avec $x_j^1 \geq 0$ et $x_j^2 \geq 0$. On obtient :

$$\begin{aligned} & \text{Maximiser } \sum_{j=1}^n (-c_j x_j^1) + \sum_{j=1}^n c_j x_j^2 \\ & \text{avec : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n (-a_{ij} x_j^1) + \sum_{j=1}^n a_{ij} x_j^2 \leq -b_i \\ \text{pour } i \in \{m+1, \dots, m+p\}, \sum_{j=1}^n (-a_{ij} x_j^1) + \sum_{j=1}^n a_{ij} x_j^2 \leq -b_i \\ \text{pour } i \in \{m+1, \dots, m+p\}, \sum_{j=1}^n a_{ij} x_j^1 + \sum_{j=1}^n (-a_{ij} x_j^2) \leq b_i \\ \text{pour } j \in \{1, 2, \dots, n\}, x_j^1 \geq 0, x_j^2 \geq 0. \end{cases} \end{aligned}$$

Le problème dual est :

$$\begin{aligned} & \text{Minimiser } \sum_{i=1}^m (-b_i y_i) - \sum_{i=m+1}^{m+p} b_i y_i^1 + \sum_{i=m+1}^{m+p} b_i y_i^2 \\ & \text{avec : } \begin{cases} \text{pour } j \in \{1, 2, \dots, n\}, \\ \sum_{i=1}^m (-a_{ij} y_i) + \sum_{i=m+1}^{m+p} (-a_{ij} y_i^1) + \sum_{i=m+1}^{m+p} a_{ij} y_i^2 \geq -c_j \\ \text{pour } j \in \{1, 2, \dots, n\}, \\ \sum_{i=1}^m a_{ij} y_i + \sum_{i=m+1}^{m+p} a_{ij} y_i^1 + \sum_{i=m+1}^{m+p} (-a_{ij} y_i^2) \geq c_j \\ \text{pour } i \in \{1, 2, \dots, m\}, y_i \geq 0, \\ \text{pour } i \in \{m+1, \dots, m+p\}, y_i^1 \geq 0, y_i^2 \geq 0. \end{cases} \end{aligned}$$

Ceci peut se réécrire :

$$\begin{aligned} & \text{Maximiser } \sum_{i=1}^m b_i y_i + \sum_{i=m+1}^{m+p} b_i (y_i^1 - y_i^2) \\ & \text{avec : } \begin{cases} \text{pour } j \in \{1, 2, \dots, n\}, \sum_{i=1}^m a_{ij} y_i + \sum_{i=m+1}^{m+p} a_{ij} (y_i^1 - y_i^2) \leq c_j \\ \text{pour } j \in \{1, 2, \dots, n\}, \sum_{i=1}^m a_{ij} y_i + \sum_{i=m+1}^{m+p} a_{ij} (y_i^1 - y_i^2) \geq c_j \\ \text{pour } i \in \{1, 2, \dots, m\}, y_i \geq 0 \\ \text{pour } i \in \{m+1, \dots, m+p\}, y_i^1 \geq 0, y_i^2 \geq 0. \end{cases} \end{aligned}$$

6.6. EXERCICES

91

En posant, pour $i \in \{m+1, \dots, m+p\}$, $y_i = y_i^1 - y_i^2$, la variable y_i est non signée et on peut encore écrire ce problème dual comme suit :

$$\begin{aligned} & \text{Maximiser } \sum_{i=1}^{m+p} b_i y_i \\ & \text{avec : } \begin{cases} \text{pour } j \in \{1, 2, \dots, n\}, \sum_{i=1}^{m+p} a_{ij} y_i = c_j \\ \text{pour } i \in \{1, 2, \dots, m\}, y_i \geq 0 \\ \text{pour } i \in \{m+1, \dots, m+p\}, y_i \in \mathbb{R}. \end{cases} \end{aligned}$$

On obtient le problème (Q) .

Q2. On utilise la question précédente en choisissant $b_i = 0$ pour $i \in \{1, \dots, m+p\}$. Les problèmes (P) et (Q) deviennent (P_0) et (Q_0) définis par :

$$\begin{aligned} & \text{Minimiser } z = \sum_{j=1}^n c_j x_j \\ (P_0) \quad & \text{avec : } \begin{cases} \text{pour } i \in \{1, 2, \dots, m\}, \sum_{j=1}^n a_{ij} x_j \geq 0 \\ \text{pour } i \in \{m+1, \dots, m+p\}, \sum_{j=1}^n a_{ij} x_j = 0 \end{cases} \end{aligned}$$

et

$$\begin{aligned} & \text{Maximiser } w = 0 \\ (Q_0) \quad & \text{avec : } \begin{cases} \text{pour } j \in \{1, 2, \dots, n\}, \sum_{i=1}^{m+p} a_{ij} y_i = c_j \\ \text{pour } i \in \{1, \dots, m\}, y_i \geq 0 \\ \text{pour } i \in \{m+1, \dots, m+p\}, y_i \in \mathbb{R}. \end{cases} \end{aligned}$$

Remarquons que l'origine est réalisable pour (P_0) (et donne à z la valeur 0).

Si la proposition (i) est vérifiée, le problème (P_0) est minoré par 0 (en fait, son minimum vaut 0). D'après le théorème de la dualité, le problème (Q_0) est réalisable, ce qui signifie que la proposition (ii) est vérifiée.

Si la proposition (ii) est vérifiée, le problème (Q_0) est réalisable, de maximum 0. D'après le théorème de la dualité, le problème (P_0) est réalisable de minimum 0, ce qui signifie que la proposition (i) est vérifiée.

Chapitre 7

Flot de valeur maximum et coupe de capacité minimum

7.1 Introduction

La recherche d'un flot de valeur maximum dans un réseau a des applications directes, par exemple l'acheminement d'une marchandise dans un réseau de transport (partie 7.6). Mais les flots en ont d'autres grâce à la modélisation de problèmes en termes de flots, ce que nous illustrerons dans la partie 7.5 par la détermination d'un couplage de cardinal maximum dans un graphe biparti, modèle par exemple de problèmes d'affectation.

L'égalité que nous établirons entre la valeur maximum d'un flot et la capacité minimum d'une coupe n'a rien de fortuit : en effet, le problème du flot de valeur maximum est un problème d'optimisation linéaire ; son problème dual admet pour solution une coupe de capacité minimum (voir l'exercice 7.3.). Dans ce chapitre, nous donnons une solution algorithmique des deux problèmes qui ne consiste pas à résoudre directement un problème linéaire par une méthode telle que l'algorithme du simplexe.

Ce chapitre comprend quatre parties. La première est consacrée aux résultats théoriques liant le problème du flot maximum à celui de la coupe de capacité minimum. La suivante présente l'algorithme de Ford et Fulkerson, qui permet de résoudre ces deux problèmes simultanément.

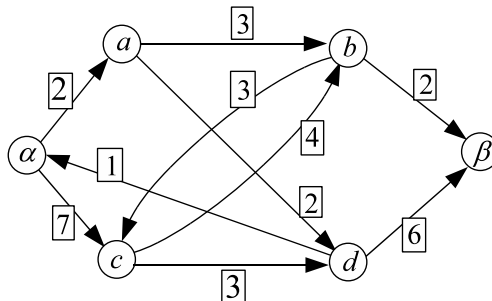


FIGURE 7.1 – Le réseau \mathcal{R}_0 .

7.2 Définitions, notations et problèmes

Définition 7.1. Dans le contexte des flots, on appelle *réseau* la donnée constituée d’un graphe¹ orienté $\mathcal{G} = (S, A)$, d’une application nommée *capacité*, notée c , définie de A dans $]0, +\infty[$, et de deux sommets particuliers, l’un appelé *source* et noté α , l’autre appelé *puits* et noté β .

On notera donc que les capacités d’un réseau ne sont jamais nulles.

La figure 7.1 représente un réseau nommé \mathcal{R}_0 de source α et de puits β ; les capacités sont les valeurs qui apparaissent, encadrées, à côté des arcs.

Définition 7.2. Soit un réseau \mathcal{R} . On appelle *flot* dans \mathcal{R} toute application f de A dans \mathbb{R} vérifiant les deux propriétés suivantes :

- pour tout arc $a \in A$, $0 \leq f(a) \leq c(a)$,
- pour tout sommet s autre que α et β , il y a conservation du flot en s , ce qui signifie que le flot total entrant en s (somme des valeurs de $f(a)$ pour tous les arcs a d’extrémité s) est égal au flot total sortant de s (somme des valeurs de $f(a)$ pour tous les arcs a d’origine s) :

$$\forall s \in S \setminus \{\alpha, \beta\}, \sum_{s' \in V^-(s)} f(s', s) = \sum_{s'' \in V^+(s)} f(s, s'').$$

On appelle *flux* d’un arc a la quantité $f(a)$ qu’il porte. Si on a $f(a) = c(a)$, on dit que l’arc a est *saturé*.

Les nombres non encadrés de la figure 7.2 définissent un flot dans le réseau \mathcal{R}_0 . Pour ce flot, l’arc (d, α) est saturé mais l’arc (c, d) ne l’est pas.

1. L’annexe A rappelle les principaux éléments de la théorie des graphes.

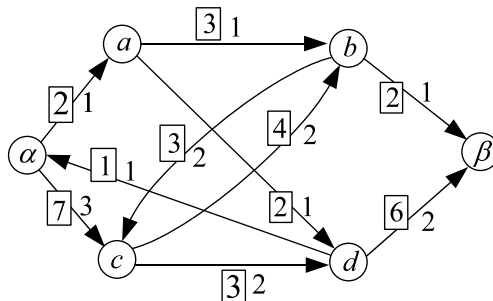


FIGURE 7.2 – Un flot dans le réseau \mathcal{R}_0 .

Définition 7.3. La *valeur d'un flot* f est égale au flot total quittant la source α moins le flot total entrant en α ; on note $val(f)$ la valeur du flot f :

$$val(f) = \sum_{s \in V^+(\alpha)} f(\alpha, s) - \sum_{s' \in V^-(\alpha)} f(s', \alpha).$$

Pour le flot de la figure 7.2, le flot sortant de α vaut $1 + 3 = 4$ alors que le flot entrant vaut 1, d'où $val(f) = 3$.

Soit \mathcal{R} un réseau. Si C est un ensemble de sommets de \mathcal{R} , on note \overline{C} le complémentaire de C dans S et on note (C, \overline{C}) l'ensemble des arcs dont l'origine est dans C et l'extrémité dans \overline{C} .

Définition 7.4. Soit \mathcal{R} un réseau. On appelle *coupe* dans \mathcal{R} tout ensemble d'arcs (C, \overline{C}) tel que C contienne α et \overline{C} contienne β .

Définition 7.5. La *capacité d'une coupe* (C, \overline{C}) , notée $cap(C, \overline{C})$, est définie par :

$$cap(C, \overline{C}) = \sum_{a \in (C, \overline{C})} c(a).$$

Pour la figure 7.1, si on prend $C_0 = \{\alpha, a, b\}$, la coupe (C_0, \overline{C}_0) contient les quatre arcs (α, c) , (a, d) , (b, c) et (b, β) et sa capacité vaut $7 + 2 + 3 + 2 = 14$.

On notera qu'une coupe et sa capacité sont définies indépendamment de tout flot. On notera aussi que la définition de la capacité d'une coupe (C, \overline{C}) ne fait pas intervenir les arcs allant de \overline{C} à C .

Ce chapitre est consacré notamment à la résolution de deux problèmes relatifs à un réseau. Le premier est la recherche d'un flot de valeur maximum. Le second est la recherche d'une coupe de capacité minimum.

Problème du flot de valeur maximum comme problème d'optimisation linéaire

La recherche d'un flot de valeur maximum s'écrit, en utilisant les notations définies plus haut :

$$\text{Maximiser} \quad \sum_{\{s \mid (\alpha, s) \in A\}} f(\alpha, s) - \sum_{\{s \mid (s, \alpha) \in A\}} f(s, \alpha)$$

avec les contraintes :

$$\begin{cases} \forall s \in S \setminus \{\alpha, \beta\}, \quad \sum_{\{t \mid (t, s) \in A\}} f(t, s) - \sum_{\{t \mid (s, t) \in A\}} f(s, t) = 0 \\ \forall (s, t) \in A, \quad 0 \leq f(s, t) \leq c(s, t). \end{cases}$$

Cette formulation établit qu'il s'agit bien d'un problème d'optimisation linéaire, ce qui permet de le résoudre à l'aide de l'algorithme du simplexe. En notant n le nombre de sommets du graphe et m le nombre d'arcs de celui-ci, le nombre de variables dans l'expression du problème sous forme standard est égal à m et le nombre de contraintes autres que les contraintes de positivité est égal à $n + m - 2$.

L'exercice 7.3. montre que le problème de la coupe de capacité minimum est le dual du problème du flot de valeur maximum au sens de l'optimisation linéaire.

7.3 Résultats théoriques

Nous allons maintenant établir des liens théoriques existant entre ces deux problèmes.

On se place toujours dans un réseau \mathcal{R} .

Proposition 7.6. Si f est un flot dans \mathcal{R} et (C, \overline{C}) une coupe dans \mathcal{R} , alors :

$$val(f) = \sum_{a \in (C, \overline{C})} f(a) - \sum_{a \in (\overline{C}, C)} f(a).$$

Preuve. Pour tout sommet s de C autre que α , d'après le principe de conservation du flot :

$$\sum_{a \text{ d'origine } s} f(a) - \sum_{a \text{ d'extrémité } s} f(a) = 0 ;$$

7.3. RÉSULTATS THÉORIQUES

97

par ailleurs :

$$\sum_{a \text{ d'origine } \alpha} f(a) - \sum_{a \text{ d'extrémité } \alpha} f(a) = val(f).$$

En faisant la somme de toutes ces égalités sur l'ensemble des sommets de C , on obtient :

$$\sum_{a \text{ d'origine dans } C} f(a) - \sum_{a \text{ d'extrémité dans } C} f(a) = val(f).$$

Pour tout arc a ayant son origine et son extrémité dans C , la quantité $f(a)$ apparaît deux fois dans l'expression précédente, une fois affectée du signe « + » dans le terme correspondant à l'origine de a , une autre fois affectée du signe « - » dans le terme correspondant à l'extrémité de a ; ces deux valeurs s'annulent donc. Par ailleurs, pour les arcs a de (C, \overline{C}) , la valeur de $f(a)$ apparaît une fois exactement, affectée du signe « + » et, pour ceux de (\overline{C}, C) , elle apparaît exactement une fois, affectée du signe « - ». D'où le résultat voulu après simplification. \diamond

Théorème 7.7. Si f est un flot dans \mathcal{R} et (C, \overline{C}) une coupe dans \mathcal{R} , alors :

1. $val(f) \leq cap(C, \overline{C})$;
2. si $val(f) = cap(C, \overline{C})$, alors f est de valeur maximum et (C, \overline{C}) est de capacité minimum ;
3. $val(f) = cap(C, \overline{C})$ si et seulement si :
 - pour tout arc a de (C, \overline{C}) , $f(a) = c(a)$;
 - pour tout arc a de (\overline{C}, C) , $f(a) = 0$.

Preuve. D'après la proposition 7.6., on a :

$$val(f) = \sum_{a \in (C, \overline{C})} f(a) - \sum_{a \in (\overline{C}, C)} f(a).$$

Par définition d'un flot :

- pour tout a dans (C, \overline{C}) , $f(a) \leq c(a)$
- pour tout a dans (\overline{C}, C) , $-f(a) \leq 0$.

En additionnant ces inégalités, on obtient le point 1.

Par ailleurs, on obtient $val(f) < cap(C, \overline{C})$ si et seulement si au moins une de ces inégalités est stricte. D'où le point 3.

Il reste le point 2. Supposons que l'on ait $val(f) = cap(C, \overline{C})$. Soit f' un autre flot dans \mathcal{R} . On a, d'après le point 1, $val(f') \leq cap(C, \overline{C})$, d'où $val(f') \leq val(f)$: le flot f est donc de valeur maximum. De même, soit (C', \overline{C}') une coupe quelconque. On a $val(f) \leq cap(C', \overline{C}')$ et donc $cap(C, \overline{C}) \leq cap(C', \overline{C}')$: la coupe (C, \overline{C}) est de capacité minimum. \diamond

Les deux problèmes du flot de valeur maximum et de la coupe de capacité minimum étant duaux l'un de l'autre (cf. exercice 7.3.), la théorie de la dualité établit la réciproque de ce théorème : si f est de valeur maximum et (C, \overline{C}) de capacité minimum, alors $val(f) = cap(C, \overline{C})$. Ce résultat sera prouvé directement plus loin, indépendamment de la théorie de la dualité.

7.4 Algorithme de Ford et Fulkerson

L'algorithme de Ford et Fulkerson (1956) offre une autre approche pour déterminer un flot de valeur maximum ainsi qu'une coupe de capacité minimum dans un réseau. Il repose sur le principe suivant : partant d'un flot dans le réseau (par exemple le flot nul), on cherche s'il existe une *chaîne augmentante* (voir ci-dessous) d'extrémités α à β permettant d'augmenter la valeur du flot ; s'il n'en existe pas, le flot est maximum ; s'il en existe une, on modifie le flot le long des arcs de cette chaîne afin d'augmenter la valeur du flot courant et on recommence avec le nouveau flot obtenu.

7.4.1 Chaîne augmentante

Supposons établi un flot f dans le réseau \mathcal{R} . On considère alors une chaîne dans \mathcal{R} , d'extrémités α et β (on rappelle que, dans une chaîne d'un graphe orienté, les arcs ne sont pas nécessairement tous dans le même sens ; voir la définition A.32., page 211). Si on veut augmenter la valeur du flot dans \mathcal{R} en modifiant $f(a)$ pour les arcs a de cette chaîne, sans modifier le flux des arcs ne faisant pas partie de la chaîne, on doit, pour assurer la conservation du flot, augmenter d'une quantité constante δ les flux $f(a)$ pour les arcs a de cette chaîne à l'endroit (c'est-à-dire les arcs dont on rencontre l'origine avant l'extrémité lorsqu'on parcourt la chaîne de α vers β) et diminuer, de

7.4. ALGORITHME DE FORD ET FULKERSON

99

cette même quantité δ , les flux $f(a)$ pour les arcs a à l'envers (c'est-à-dire les arcs dont on rencontre l'extrémité avant l'origine lorsqu'on parcourt la chaîne de α vers β). Considérons en effet deux arcs a et b consécutifs sur la chaîne, incidents en s et, par exemple, parcourus à l'endroit. Si on augmente le flot d'une valeur δ sur l'arc a , pour continuer à assurer la conservation du flot en s , sous l'hypothèse qu'elle l'était avant cette modification, on voit que l'on est amené à l'augmenter de la même quantité sur l'arc b : la quantité excédentaire δ qui entre en s doit en effet quitter s . Les autres cas se traitent de manière analogue (voir aussi la figure 7.5).

Plus précisément, étant donné un flot f , on définit une *chaîne augmentante* pour f de α à un sommet quelconque s de la manière suivante.

Définition 7.8. Soit f un flot. Soit Ch une chaîne entre α et un sommet quelconque s . On pose :

$$\delta^+(Ch) = \min\{c(a) - f(a) \text{ pour } a \text{ arc de } Ch \text{ parcouru à l'endroit}\}$$

$$\delta^-(Ch) = \min\{f(a) \text{ pour } a \text{ arc de } Ch \text{ parcouru à l'envers}\}$$

$$\delta(Ch) = \min\{\delta^+(Ch), \delta^-(Ch)\}.$$

On dit que Ch est une *chaîne augmentante* pour f de α à s si on a $\delta(Ch) > 0$. On dit alors que Ch est augmentante de $\delta(Ch)$.

Ayant déterminé, s'il en existe, une chaîne augmentante Ch de α à β d'une quantité $\delta > 0$, on augmente le flux de δ pour les arcs de Ch à l'endroit et on le diminue de δ pour les arcs de Ch à l'envers : on obtient ainsi un nouveau flot f' vérifiant $val(f') = val(f) + \delta$.

La figure 7.3 propose quatre exemples de chaînes de α au sommet s , augmentantes ou non augmentantes, extraites d'un réseau. Les capacités sont encadrées, les flux ne le sont pas.

La chaîne $Ch1$ est augmentante : les trois arcs à l'endroit, (α, a) , (a, b) et (c, s) , donnent $\delta^+(Ch1) = \min\{13 - 8, 10 - 7, 12 - 6\} = 3$, tandis que l'arc à l'envers, (c, b) , donne $\delta^-(Ch1) = 4$, d'où $\delta(Ch1) = \min\{3, 4\} = 3$; $Ch1$ est bien augmentante de α à s , de 3. Si on a $s = \beta$, on peut augmenter la valeur du flot de 3 : le flux de l'arc (α, a) passe à $8 + 3 = 11$, le flux de l'arc (a, b) passe à $7 + 3 = 10$, le flux de l'arc (c, b) passe à $4 - 3 = 1$, le flux de l'arc (c, β) passe à $6 + 3 = 9$.

La chaîne $Ch2$ est augmentante de α à s , de 2. L'arc le plus limitant est l'arc (à l'envers) (c, b) sur lequel la diminution est au maximum de 2. Si on a $s = \beta$, on peut augmenter la valeur du flot de 2 : le flux de l'arc (α, a) passe à $8 + 2 = 10$, le flux de l'arc (a, b) passe à $7 + 2 = 9$, le flux de l'arc (c, b) passe à $2 - 2 = 0$, le flux de l'arc (c, β) passe à $6 + 2 = 8$.

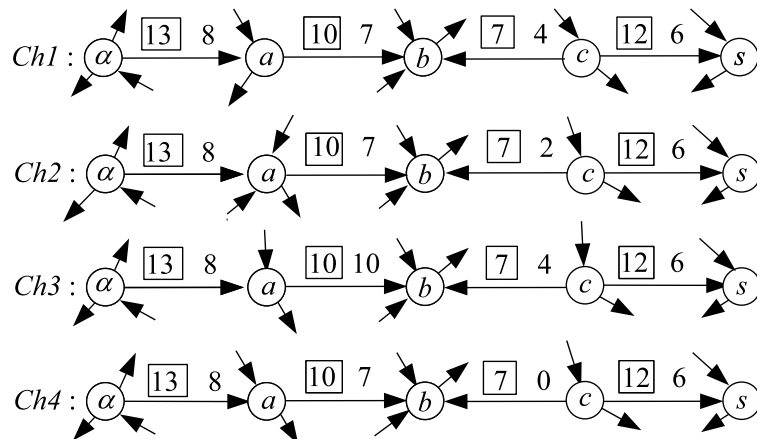


FIGURE 7.3 – Chaînes augmentantes ($Ch1$ avec $\delta(Ch1) = 3$, $Ch2$ avec $\delta(Ch2) = 2$) ou non ($Ch3$, $Ch4$) de α à s .

La chaîne $Ch3$ n'est pas augmentante de α à s car l'arc (à l'endroit) (a, b) est saturé : son flux ne peut pas être augmenté.

La chaîne $Ch4$ n'est pas augmentante de α à s car le flux de l'arc (à l'envers) (c, b) est nul et ne peut donc pas être diminué.

Proposition 7.9. Soit f un flot dans un réseau. Le flot f est de valeur maximum si et seulement si il n'existe pas de chaîne augmentante de α à β .

Preuve. L'existence d'une chaîne augmentante de α à β implique clairement que le flot n'est pas de valeur maximum.

Démontrons la réciproque. Supposons qu'il n'existe pas de chaîne augmentante de α à β . Appelons C l'ensemble des sommets s pour lesquels il existe une chaîne augmentante de α à s (on considérera que α fait partie d'un tel ensemble, en posant conventionnellement qu'une chaîne réduite à α est augmentante de α à α). Le sommet α est dans C et, par hypothèse, on suppose que β n'est pas dans C ; par conséquent, (C, \overline{C}) est une coupe.

Soit $a = (s, t)$ un arc dont l'origine s est dans C et l'extrémité t dans \overline{C} . Il existe une chaîne augmentante de α à s mais cette chaîne ne peut pas être prolongée en une chaîne augmentante de α à t ; c'est donc que l'arc a est saturé : $f(a) = c(a)$ (voir la figure 7.4).

Soit $a' = (s', t')$ un arc dont l'origine s' est dans \overline{C} et l'extrémité t' dans C . Il existe une chaîne augmentante de α à t' , mais cette chaîne ne peut pas être

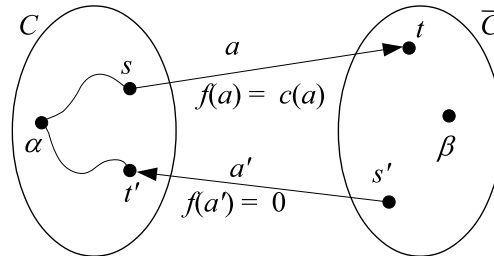


FIGURE 7.4 – Illustration pour la preuve de la proposition 7.9.

prolongée en une chaîne augmentante de α à s' ; c'est que sur l'arc a' circule un flux nul : $f(a') = 0$ (voir la figure 7.4).

D'après le théorème 7.7., on déduit que f est de valeur maximum et que la coupe (C, \overline{C}) est de capacité minimum. \diamond

On peut maintenant énoncer le théorème suivant :

Théorème 7.10. (Théorème du flot et de la coupe) Soit \mathcal{R} un réseau ; la valeur maximum d'un flot dans \mathcal{R} est égale à la capacité minimum d'une coupe dans \mathcal{R} .

Preuve. Nous avons remarqué que le problème de la recherche du flot maximum est un problème d'optimisation linéaire ; de plus ce problème est réalisable puisque le flot nul est une solution réalisable et il est borné par la valeur d'une coupe quelconque. En conséquence, le problème atteint son maximum.

Considérons un flot f de valeur maximum. Il n'existe pas pour f de chaîne augmentante de α à β . En reprenant la démonstration de la proposition 7.9., la coupe (C, \overline{C}) obtenue en séparant tous les sommets tels qu'il existe une chaîne augmentante depuis α (dont α) des autres sommets (dont β) a une capacité égale à la valeur du flot. Le théorème 7.7. permet de conclure. \diamond

7.4.2 Description de l'algorithme de Ford et Fulkerson

Le principe de l'algorithme de Ford et Fulkerson est le suivant :

- partir d'un flot dans le réseau (par exemple, le flot nul)
- tant qu'il existe une chaîne augmentante pour le flot courant de α à β , augmenter la valeur du flot en utilisant cette chaîne.

Plus précisément, soit f un flot dans le réseau. On appelle *sommet marqué* soit le sommet α , soit un sommet tel qu'on a exhibé une chaîne augmentante de α à ce sommet pour le flot f . On attribue à chaque sommet marqué t une *marque* à deux places : α est marqué par $(., .)$; pour $t \neq \alpha$, la première place de la marque de t mémorise le nom du sommet qui permet de marquer t , c'est-à-dire du sommet s dont on peut prolonger la chaîne augmentante depuis α jusqu'à t ; dans la seconde place de la marque de t , si t a été marqué par s :

- si l'arc utilisé était (s, t) (c'est-à-dire un arc à l'endroit par rapport à la chaîne augmentante), on met le signe « $+$ »,
- si l'arc utilisé était (t, s) (c'est-à-dire un arc à l'envers par rapport à la chaîne augmentante), on met le signe « $-$ ».

Ce signe est indispensable quand le graphe n'est pas antisymétrique, afin de savoir si un sommet t marqué par un sommet s l'est en tant que successeur ou prédécesseur de s , autrement dit, afin de savoir si c'est l'arc (s, t) ou l'arc (t, s) qui figure dans la chaîne augmentante en cours de construction.

Examiner un sommet marqué s consiste à déterminer si l'on peut prolonger la chaîne augmentante de α à s jusqu'à un nouveau sommet successeur ou prédécesseur de s non encore marqué.

On définit alors un algorithme de recherche d'une chaîne augmentante, dit *algorithme de marquage*, applicable à tout flot f donné. Pour initialiser cet algorithme, on considère que seul le sommet α est marqué ; aucun sommet n'est examiné. L'algorithme de recherche d'une chaîne augmentante consiste, tant que le puits β n'est pas marqué et qu'il existe un sommet marqué et non examiné, à examiner un tel sommet. Plus précisément, pour le flot courant f :

- considérer α comme marqué par $(., .)$;
- pour tout sommet $s \neq \alpha$, considérer s comme non marqué ;
- considérer qu'aucun sommet n'est examiné ;
- tant que β est non marqué et qu'il existe un sommet marqué non examiné, faire
 - ★ soit s un sommet marqué non examiné ;
 - ★ pour tout successeur t de s qui n'est pas marqué et vérifiant $f(s, t) < c(s, t)$ (l'arc (s, t) n'est pas saturé), marquer t par $(s, +)$;
 - ★ pour tout prédécesseur t de s qui n'est pas marqué et vérifiant $f(t, s) > 0$ (l'arc (t, s) n'est pas de flux nul), marquer t par $(s, -)$;
 - ★ considérer s comme examiné.

7.4. ALGORITHME DE FORD ET FULKERSON

103

À la fin de l'algorithme de marquage, si le puits est marqué, une chaîne augmentante Ch est mise en évidence ; on peut la reconstituer de proche en proche, en partant de β , à l'aide des indications conservées dans les marques. On utilise alors Ch pour améliorer le flot courant. Pour cela, il faut d'abord calculer la quantité dont Ch est augmentante ; on utilise la procédure suivante, dans laquelle on appelle *marque1* et *marque2* les deux champs des marques :

- $t \leftarrow \beta$
- $\delta \leftarrow \infty$
- tant que $t \neq \alpha$
 - ★ $s \leftarrow \text{marque1}(t)$
 - ★ si $\text{marque2}(t) = \llcorner + \rceil$, $\delta \leftarrow \min\{\delta, c(s, t) - f(s, t)\}$
sinon $\delta \leftarrow \min\{\delta, f(t, s)\}$
 - ★ $t \leftarrow s$.

Puis on utilise la quantité calculée δ pour augmenter la valeur du flot avec cette seconde procédure :

- $t \leftarrow \beta$
- tant que $t \neq \alpha$
 - ★ $s \leftarrow \text{marque1}(t)$
 - ★ si $\text{marque2}(t) = \llcorner + \rceil$, $f(s, t) \leftarrow f(s, t) + \delta$
sinon $f(t, s) \leftarrow f(t, s) - \delta$
 - ★ $t \leftarrow s$.

La structure obtenue à la fin de cette seconde procédure est bien un flot. En effet, grâce aux opérateurs « min » appliqués pour le calcul de δ , la valeur de δ est telle que, après la mise à jour des flux, ceux-ci sont encore compris entre 0 et la capacité des arcs. En outre, la loi de conservation est bien satisfaite après cette mise à jour. Pour s'en convaincre, considérons un sommet s par lequel passe la chaîne augmentante et les deux arcs de cette chaîne passant par s . Selon l'orientation de ces deux arcs quand on parcourt la chaîne de α vers β , la variation des flux de ces arcs permet de conserver le fait que la somme de ce qui entre en s reste égale à la somme de ce qui en sort : comme cela était vrai avant la mise à jour, il suffit de vérifier que ceci est satisfait par les variations des flux de ces deux arcs ; la figure 7.5 montre que tel est bien le cas.

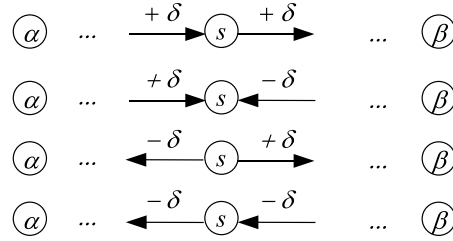


FIGURE 7.5 – Conservation du flot en un sommet s d’une chaîne augmentante selon l’orientation des deux arcs de cette chaîne incidents à s .

Augmenter le flot en utilisant une chaîne augmentante consiste à effectuer les deux procédures décrites ci-dessus : déterminer de combien cette chaîne permet d’augmenter le flot courant f puis modifier f pour en augmenter la valeur. Si l’algorithme de recherche d’une chaîne augmentante n’a pas marqué le puits, il n’existe pas de chaîne augmentante de α à β et le flot courant est maximum. Ce qui donne la description suivante de l’algorithme de Ford et Fulkerson :

- définir un flot f (éventuellement le flot nul)
- répéter
 - ★ appliquer l’algorithme de marquage
 - ★ si β est marqué, alors augmenter le flot f en utilisant la chaîne augmentante de α à β déterminée par l’algorithme de marquage jusqu’à ce que β ne soit plus marqué.

À la sortie de cette boucle, si on en sort (voir ci-dessous), le flot est de valeur maximum, comme nous allons maintenant le prouver.

7.4.3 Preuve, convergence et complexité de l’algorithme

Si on sort de la boucle « répéter », l’algorithme de marquage s’est terminé sans avoir marqué β . Soit M l’ensemble des sommets qui ont été marqués au cours du dernier marquage. L’ensemble M contient α (puisque toute itération de l’algorithme de marquage commence par marquer α), alors que β ne lui n’appartient pas : (M, \overline{M}) est donc une coupe. Soit $a = (s, t)$ un arc de (M, \overline{M}) ; le sommet s est marqué et donc examiné ; lors de l’examen de s , le sommet t n’a pas été marqué puisqu’il est dans \overline{M} ; c’est donc que l’arc a est saturé. Soit $a' = (s', t')$ un arc de (\overline{M}, M) ; le sommet t' est marqué et donc

7.4. ALGORITHME DE FORD ET FULKERSON

105

examiné ; lors de l'examen de t' , le sommet s' n'a pas été marqué puisqu'il est dans \overline{M} ; c'est donc que l'arc a' porte un flux nul. Par application du théorème 7.7., la capacité de la coupe (M, \overline{M}) est égale à la valeur du flot f , le flot est de valeur maximum et la coupe est de capacité minimum.

On obtient ainsi la proposition suivante :

Proposition 7.11. Si on note M l'ensemble des sommets marqués pendant la dernière application de l'algorithme de marquage, l'ensemble (M, \overline{M}) est une coupe de capacité minimum.

Pour étudier la convergence et la complexité de l'algorithme de Ford et Fulkerson, supposons d'abord que toutes les capacités sont des entiers. Partant du flot nul (ou en fait de tout flot dont les flux sont entiers), à toute itération autre que la dernière (celle qui ne marque pas β et ne conduit donc pas à une chaîne augmentante) on augmente la valeur du flot d'une quantité entière strictement positive ; or, on sait que la valeur maximum du flot est bornée par la capacité de toute coupe séparant la source du puits. On est donc assuré d'avoir un nombre fini de recherches de chaînes augmentantes. Dès que les capacités sont entières, il y a donc convergence en un nombre fini d'itérations.

Si les capacités sont rationnelles, on peut les rendre entières en les multipliant par le plus petit commun multiple P des dénominateurs pour les rendre entières. Le nombre d'étapes est alors majoré par la capacité d'une coupe quelconque multipliée par P .

En revanche, dans le cas de capacités réelles non rationnelles, l'exercice 7.2. met en évidence un exemple où l'algorithme de Ford et Fulkerson ne converge pas en un nombre fini d'étapes et, avec un nombre infini d'étapes, converge vers une valeur qui n'est pas maximum.

Lorsque l'on est dans le cas où toutes les capacités sont entières, le nombre de phases de l'algorithme est majoré par la valeur du flot, et donc par la capacité de toute coupe, par exemple par celle définie par $C = \{\alpha\}$. Par ailleurs, au cours d'une itération, on examine au plus tous les sommets autres que β , c'est-à-dire $n - 1$ sommets si l'ordre du graphe est n . Or, examiner un sommet consiste à effectuer au plus un nombre d'opérations élémentaires proportionnel à la somme du degré extérieur et du degré intérieur de ce sommet, pour voir si l'on peut prolonger la chaîne augmentante arrivant à ce sommet. Le nombre d'opérations élémentaires à effectuer pour la recherche

d’une chaîne augmentante est donc borné par la somme des degrés extérieurs et intérieurs des sommets, c’est-à-dire deux fois le nombre d’arcs m (ceci requiert une structure de données appropriée pour décrire le graphe : c’est le cas en supposant le graphe codé à la fois par les listes de successeurs et les listes de prédécesseurs). La complexité de l’algorithme de Ford et Fulkerson est donc :

- $O(m \times val_{max})$ où val_{max} désigne la valeur maximum du flot,
- ou aussi $O(m \times cap(C, \overline{C}))$ où (C, \overline{C}) est une coupe quelconque,
- ou encore $O(m \times n \times c_{max})$ où c_{max} désigne la capacité maximum des arcs, ce dernier majorant s’obtenant par exemple en considérant la coupe définie par $C = \{\alpha\}$, de capacité majorée par $n \times c_{max}$.

Notons encore un résultat lié à l’intégrité des capacités :

Théorème 7.12. Dans un réseau à capacités entières, il existe un flot maximum tel que tous les flux soient entiers.

Preuve. Appliquons l’algorithme de Ford et Fulkerson à partir d’un flot dont les flux sont entiers (par exemple le flot nul). On constate que la procédure décrite plus haut pour déterminer la quantité δ dont on va augmenter la valeur du flot courant produit un entier. En effet, si on suppose qu’à l’itération courante les flux $f(s, t)$ sont des entiers, les capacités étant elles-mêmes supposées entières, le calcul de δ ne fait intervenir que des opérations (la soustraction, le minimum) qui, appliquées à des entiers, donnent aussi un entier. De même, la procédure qui met à jour le flot courant ne fait intervenir que des opérations (l’addition, la soustraction) qui, appliquées à des entiers, donnent aussi un entier. Ainsi, si le flot considéré avant cette mise à jour n’a que des flux entiers, il en est de même après la mise à jour. Une récurrence immédiate sur les itérations de l’algorithme montre donc que, si on part d’un flot dont les flux sont entiers (ce que l’on peut toujours faire en partant du flot nul), le flot obtenu à chaque itération de l’algorithme est lui aussi à flux entiers, en particulier à la dernière itération où le flot est maximum. \diamond

Bien sûr, ce qui précède n’exclut pas l’existence d’un flot maximum dont les flux ne sont pas des entiers alors que les capacités sont entières, ni l’existence d’un flot maximum dont les flux sont des entiers alors que les capacités ne sont pas toutes entières.

7.4. ALGORITHME DE FORD ET FULKERSON

107

Enfin, signalons une variante, due à J. Edmonds et R. Karp, portant sur la façon d’examiner les sommets marqués : si on cherche les chaînes augmentantes par un parcours en largeur (obtenu en gérant les sommets marqués qui restent à examiner selon une file), le nombre d’itérations de l’algorithme ne dépasse pas $n \times m$, résultat que nous ne justifions pas ici. L’algorithme converge alors toujours vers un flot de valeur maximum, avec une complexité en $O(n \times m^2)$, quelles que soient les capacités.

7.4.4 Exemple d’application de l’algorithme de Ford et Fulkerson

On considère le réseau \mathcal{R}_0 représenté par la figure 7.1 et on cherche un flot de valeur maximum à l’aide de l’algorithme de Ford et Fulkerson ; on initialise le flot avec le flot nul, représenté sur la figure 7.6 où les nombres encadrés, à côté des arcs, indiquent les capacités, tandis que les autres indiquent les flux courants.

Dans tout l’algorithme, on fait le choix d’examiner les sommets marqués non examinés selon une file (variante d’Edmonds et Karp).

On effectue une première étape de l’algorithme de Ford et Fulkerson en appliquant l’algorithme de marquage. On marque d’abord uniquement le sommet α . On examine le seul sommet marqué et non examiné, c’est-à-dire α ; on marque alors les sommets a et c avec la marque $(\alpha, +)$. On examine ensuite le sommet a qui permet de marquer les sommets b et d avec la marque $(a, +)$. On examine le sommet c dont tous les voisins sont déjà marqués : on ne marque rien. On examine le sommet b qui permet de marquer le sommet β avec la marque $(b, +)$. Le sommet β étant marqué, l’algorithme de marquage s’arrête : on a trouvé une chaîne augmentante, la chaîne $\alpha \rightarrow a \rightarrow b \rightarrow \beta$. Cette chaîne est augmentante de $\delta = 2$; on actualise le flot en conséquence et on obtient le flot indiqué dans la figure 7.6 à l’étape 1 (à droite).

On effectue l’étape 2. On applique l’algorithme de marquage. On marque le sommet α que l’on examine juste ensuite ; l’arc (α, a) étant saturé, on ne peut pas marquer le sommet a ; on marque le sommet c avec la marque $(\alpha, +)$. On examine ensuite le sommet c qui permet de marquer les sommets b et d avec la marque $(c, +)$. On examine le sommet b ; l’arc (b, β) étant saturé, on ne peut pas marquer le sommet β par b ; l’arc (a, b) portant un flux non nul, on marque le sommet a avec la marque $(b, -)$. On examine le sommet d qui permet de marquer le sommet β avec la marque $(d, +)$. Le sommet β

étant marqué, l'algorithme de marquage s'arrête et on a trouvé une chaîne augmentante, la chaîne $\alpha \rightarrow c \rightarrow d \rightarrow \beta$. Cette chaîne est augmentante de 3, on actualise le flot et on obtient le flot indiqué dans la figure 7.6 à l'étape 2 (à droite). On peut synthétiser le résultat de l'algorithme de marquage à cette étape par le tableau ci-dessous.

sommets examinés	sommets marqués et marques
α	$c (\alpha, +)$
c	$b (c, +)$ et $d (c, +)$
b	$a (b, -)$
d	$\beta (d, +)$

On effectue l'étape 3. On applique l'algorithme de marquage. Le déroulement de l'algorithme est représenté par le tableau ci-dessous.

sommets examinés	sommets marqués et marques
α	$c (\alpha, +)$
c	$b (c, +)$
b	$a (b, -)$
a	$d (a, +)$
d	$\beta (d, +)$

Le sommet β étant marqué, l'algorithme de marquage s'arrête et on a trouvé une chaîne augmentante, la chaîne $\alpha \rightarrow c \rightarrow b \leftarrow a \rightarrow d \rightarrow \beta$. Cette chaîne est augmentante de 2, on actualise le flot et on obtient le flot indiqué dans la figure 7.6 à l'étape 3 (à droite). On effectue l'étape 4. On applique l'algorithme de marquage. Le déroulement de l'algorithme est représenté par le tableau ci-dessous.

sommets examinés	sommets marqués et marques
α	$c (\alpha, +)$
c	$b (c, +)$
b	

Le sommet β n'a pas été marqué et tout sommet marqué a été examiné, l'algorithme est terminé. Le flot est de valeur 7 et c'est la valeur maximum d'un flot dans le réseau. En notant M l'ensemble des sommets marqués pendant cette dernière étape, c'est-à-dire ici $M = \{\alpha, b, c\}$, (M, \overline{M}) est une coupe

7.4. ALGORITHME DE FORD ET FULKERSON

109

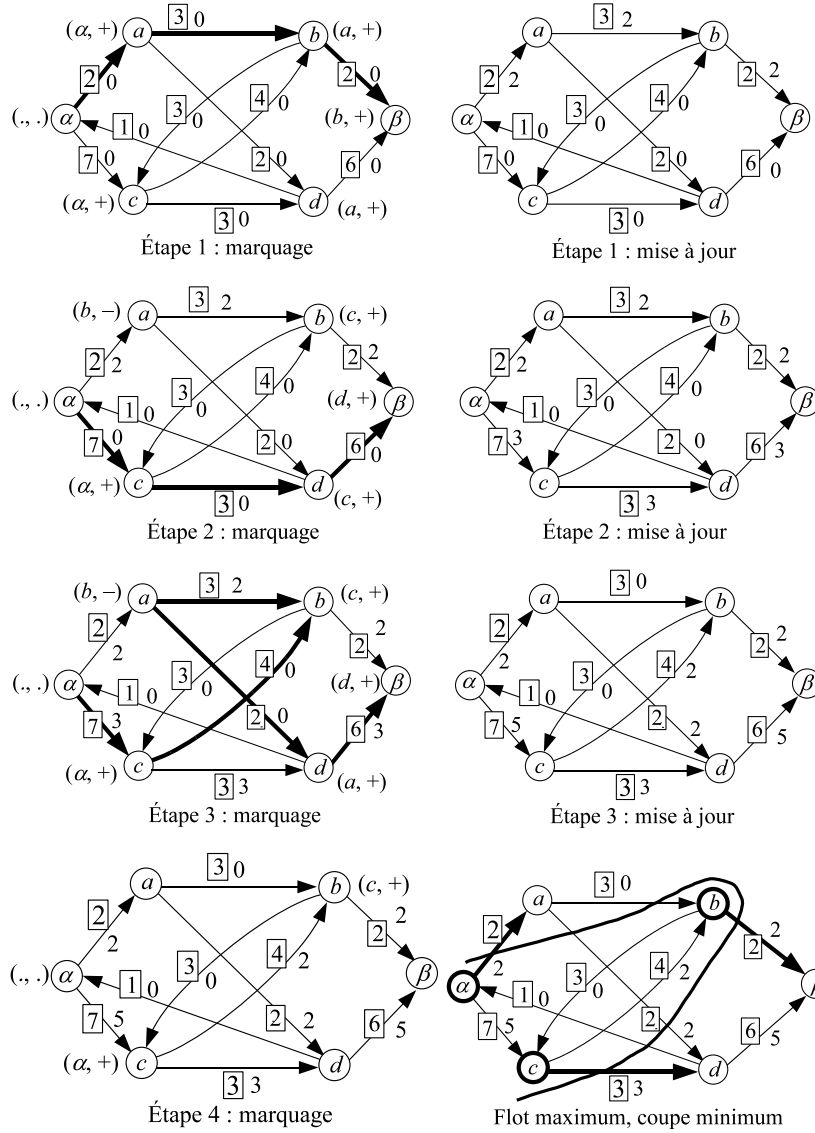


FIGURE 7.6 – Déroulement de l'algorithme de Ford et Fulkerson.

de capacité minimum. Dans le graphe représenté sur la figure 7.6 à droite de l'étape 4, on a tracé en gras les sommets de M et les arcs de la coupe correspondante ; la capacité de cette coupe vaut bien 7, comme la valeur du flot.

On peut aussi vérifier que les arcs de (M, \overline{M}) sont saturés alors que les arcs de (\overline{M}, M) sont de flux nul.

7.5 Application à un problème d’affectation

7.5.1 Un exemple et quelques définitions

On cherche à réaliser six tâches $t_1, t_2, t_3, t_4, t_5, t_6$ qui peuvent être exécutées sur une des cinq machines m_1, m_2, m_3, m_4, m_5 comme l’indiquent les étoiles dans le tableau suivant.

	m_1	m_2	m_3	m_4	m_5
t_1	★		★		
t_2	★	★	★		
t_3	★			★	
t_4	★			★	
t_5		★			★
t_6				★	★

Une machine ne peut exécuter qu’une seule tâche. On cherche à affecter les tâches aux machines de façon à réaliser le plus possible de tâches (en utilisant chaque machine une seule fois).

Définition 7.13. Dans un graphe non orienté, on appelle *couplage* un ensemble d’arêtes tel que deux quelconques de ces arêtes n’aient pas d’extrémité commune.

On construit un graphe biparti $\mathcal{G} = (S, A)$ dont les sommets représentent les tâches et les machines et où une arête lie deux sommets représentant une tâche t et une machine m si la tâche t peut être réalisée par la machine m (voir la figure 7.7). Aucune arête ne lie deux tâches ou deux machines : le graphe est biparti. Chercher à exécuter un maximum de tâches revient à chercher parmi les arêtes de ce graphe un couplage de cardinal maximum.

7.5.2 Modélisation d’un problème d’affectation

Le problème évoqué dans la section précédente est un cas particulier du *problème d’affectation* ; il se traduit par la recherche d’un couplage maximum dans un graphe biparti.

7.5. APPLICATION À UN PROBLÈME D’AFFECTATION

111

On considère donc un graphe biparti $\mathcal{G} = (S, A)$; on note U et V les deux parties des sommets. On cherche dans \mathcal{G} un couplage de cardinal maximum. Pour résoudre ce problème, nous allons nous ramener à la recherche d’un flot maximum dans un réseau \mathcal{R} que nous définissons à partir du graphe \mathcal{G} :

- on oriente les arêtes pour en faire des arcs allant de U vers V ;
- on ajoute deux sommets, α et β , qui sont la source et le puits de \mathcal{R} ;
- on ajoute les arcs (α, u) pour u appartenant à U ;
- on ajoute les arcs (v, β) pour v appartenant à V ;
- on attribue une capacité de 1 à tous les arcs.

Pour l’exemple des tâches à réaliser vu précédemment, le réseau \mathcal{R} obtenu est illustré par la figure 7.8.

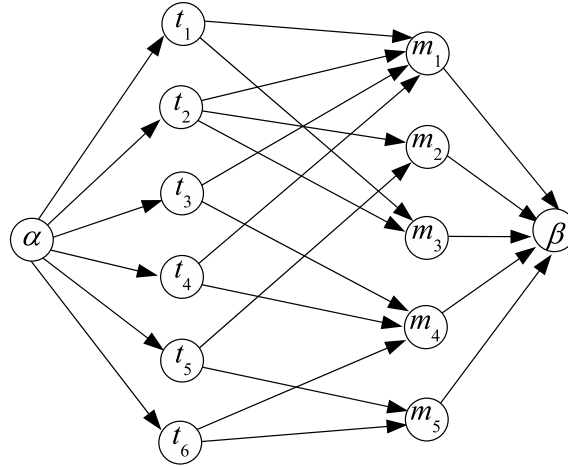
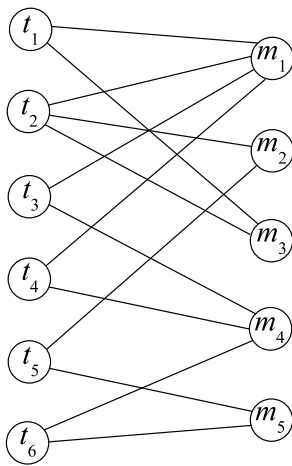


FIGURE 7.7 – Le graphe \mathcal{G} .

FIGURE 7.8 – Le réseau \mathcal{R} .

On va établir une bijection ϕ entre les flots à flux entiers dans \mathcal{R} et les couplages dans \mathcal{G} .

Soit f un flot à flux entiers dans \mathcal{R} . Les flux sur les arcs sont tous de valeur 0 ou 1. Grâce au choix des capacités unitaires sur les arcs d’origine α ou d’extrémité β et à la conservation des flots, les arcs entre U et V portant un flux de 1 sont entre eux d’origines et d’extrémités distinctes. On associe à ce flot l’ensemble des arêtes de \mathcal{G} correspondant aux arcs entre U et V portant un flux de 1 ; il s’agit d’un couplage. On note \mathcal{C} ce couplage et on

pose $\phi(f) = \mathcal{C}$. Cette application ϕ est clairement injective. De plus, soit \mathcal{C} un couplage dans \mathcal{G} . Définissons un flot f à flux entiers dans \mathcal{R} en mettant un flux de 1 sur les arcs correspondant aux arêtes de \mathcal{C} , la valeur 0 aux autres arcs qui vont de U à V et en complétant de façon à assurer la conservation du flot ; cela est possible puisqu’il n’y a pas deux arcs issus d’un même sommet de U portant un flux de 1 ni deux arcs d’extrémité dans V portant un flux de 1. On a par construction : $\phi(f) = \mathcal{C}$. L’application ϕ est donc surjective. C’est bien une bijection.

Par ailleurs, pour tout flot f à flux entiers dans \mathcal{R} , on a :

$$val(f) = \text{cardinal}(\phi(f)).$$

Si f est un flot de valeur maximum dans \mathcal{R} , $\phi(f)$ est un couplage de cardinal maximum dans \mathcal{G} et réciproquement. Chercher un couplage de cardinal maximum dans \mathcal{G} revient à chercher un flot à flux entiers de valeur maximum dans \mathcal{R} .

Notons p le cardinal de U et q celui de V . Supposons qu’on applique l’algorithme de Ford et Fulkerson pour résoudre le problème. Le nombre de recherches de chaînes augmentantes est majoré par $\min\{p, q\}$ (majorant de la valeur maximum du flot) et chaque recherche de chaîne augmentante a une complexité en $O(pq)$. La complexité de la résolution est ainsi en $O(pq \min\{p, q\})$.

7.5.3 Traitement de l’exemple de la partie 7.5.1

On applique l’algorithme de Ford et Fulkerson dans le réseau représenté par la figure 7.8 à partir du flot nul. En examinant les sommets marqués selon une file (variante d’Edmonds et Karp), on obtient les marquages suivants.

sommets examinés	sommets marqués et marques
α	$t_1 (\alpha, +), t_2 (\alpha, +), t_3 (\alpha, +), t_4 (\alpha, +),$ $t_5 (\alpha, +), t_6 (\alpha, +)$
t_1	$m_1 (t_1, +), m_3 (t_1, +)$
t_2	$m_2 (t_2, +)$
t_3	$m_4 (t_3, +)$
t_4	
t_5	$m_5 (t_5, +)$
t_6	
m_1	$\beta (m_1, +)$

7.5. APPLICATION À UN PROBLÈME D’AFFECTATION

113

On a ainsi trouvé la chaîne augmentante $\alpha \rightarrow t_1 \rightarrow m_1 \rightarrow \beta$. Après avoir actualisé le flot, on recommence et on obtient de même, successivement, les chaînes :

$$\alpha \rightarrow t_2 \rightarrow m_2 \rightarrow \beta,$$

$$\alpha \rightarrow t_3 \rightarrow m_4 \rightarrow \beta,$$

$$\alpha \rightarrow t_5 \rightarrow m_5 \rightarrow \beta.$$

Le flot obtenu est représenté sur la figure 7.9.

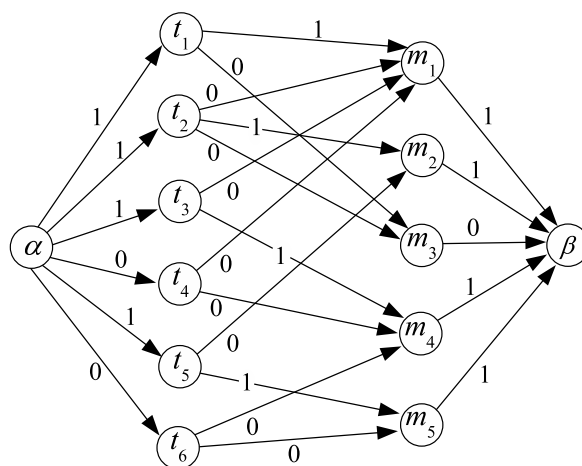


FIGURE 7.9 – Flot après quatre chaînes augmentantes.

On cherche à nouveau une chaîne augmentante et on obtient les marquages suivants :

sommets examinés	sommets marqués et marques
α	$t_4 (\alpha, +), t_6 (\alpha, +)$
t_4	$m_1 (t_4, +), m_4 (t_4, +)$
t_6	$m_5 (t_6, +)$
m_1	$t_1 (m_1, -)$
m_4	$t_3 (m_4, -)$
m_5	$t_5 (m_5, -)$
t_1	$m_3 (t_1, +)$
t_3	
t_5	$m_2 (t_5, +)$
m_3	$\beta (m_3, +)$

Le puits a été marqué. On a trouvé une chaîne augmentante qu'on reconstitue en reculant à partir de $\beta : \alpha \rightarrow t_4 \rightarrow m_1 \leftarrow t_1 \rightarrow m_3 \rightarrow \beta$; en exploitant cette chaîne augmentante, on obtient le flot représenté sur la figure 7.10.

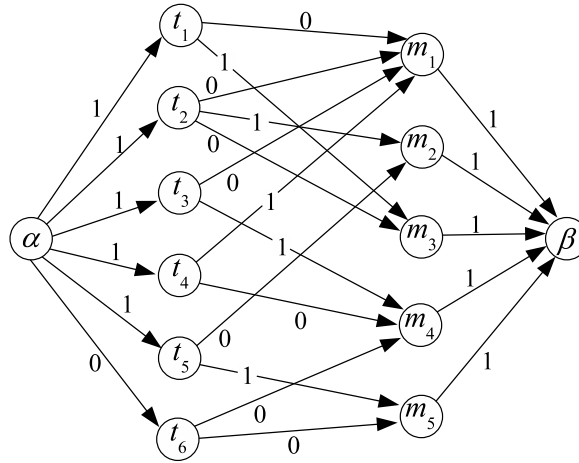


FIGURE 7.10 – Flot après cinq chaînes augmentantes.

Tous les arcs d'extrémité β étant saturés, le flot est maintenant de valeur maximum². Un couplage de cardinal maximum dans \mathcal{G} correspond aux arcs allant des tâches vers les machines portant un flux de 1; il s'agit du couplage constitué des arêtes : $\{t_1, m_3\}, \{t_2, m_2\}, \{t_3, m_4\}, \{t_4, m_1\}, \{t_5, m_5\}$. Ce couplage donne l'affectation des tâches sur les machines; la tâche t_6 n'est pas exécutée dans cette affectation.

L'exercice 7.4. propose une variante de ce problème d'affectation.

7.6 Application à un problème de transport

Un ensemble de huit entrepôts, appelés a, b, c, d, e, f, g et h , disposent d'une même marchandise. Compte tenu des problèmes de place, les entrepôts a et b ont une quantité excédentaire de marchandise; plus précisément :

2. Cet argument n'est pas toujours applicable, contrairement à celui de la coupe obtenue à l'aide des sommets marqués lors de l'itération qui n'atteint pas le puits. Pour cet exemple, une dernière application de l'algorithme de marquage permet de marquer les sommets $\alpha, t_6, m_4, m_5, t_3, t_5, m_1, m_4, m_2, t_4, t_3, t_2, m_3, t_1$.

7.6. APPLICATION À UN PROBLÈME DE TRANSPORT

115

- l'entrepôt a possède 12 unités de marchandise en trop ;
- l'entrepôt b possède 8 unités de marchandise en trop.

Compte tenu des commandes, les entrepôts f , g et h sont déficitaires ; plus précisément :

- pour l'entrepôt f , il manque 9 unités de marchandise ;
- pour l'entrepôt g , il manque 6 unités de marchandise ;
- pour l'entrepôt h , il manque 4 unités de marchandise ;

Les entrepôts c , d et e ne sont ni déficitaires, ni excédentaires.

Des camions peuvent aller d'un entrepôt à un autre entrepôt ; sur le graphe G représenté sur la figure 7.11, un arc est tracé d'un entrepôt i à un entrepôt j s'il y a possibilité d'envoyer un camion de i vers j ; pour i et j fixés, un camion ne peut faire le trajet de i vers j qu'une seule fois. Chaque camion a une certaine capacité de transport : un camion ne pourra pas transporter plus d'unités de marchandise que sa capacité. Un entrepôt excédentaire ne doit pas avoir davantage de marchandises après les opérations de transport et ne doit pas non plus être devenu déficitaire. Un entrepôt déficitaire ne doit pas avoir moins de marchandises après les opérations de transport et ne doit pas non plus être devenu excédentaire. Un entrepôt ni déficitaire, ni excédentaire doit le rester. Sur le graphe \mathcal{G} , la capacité de chaque camion est indiquée à côté de l'arc correspondant en écriture droite. Un plan de transport est provisoirement envisagé ; les quantités que l'on envisage de transporter sont marquées en italique à côté des capacités des arcs correspondants.

On note Δ la différence entre la somme des excédents des entrepôts a et b avant le transport et la somme de ces excédents après le transport. Il s'agit d'établir un plan de transport de façon à maximiser la valeur de Δ . Pour le plan de transport proposé, l'excédent du sommet a passe de 12 à 3 et l'excédent du sommet b passe de 8 à 3 ; la valeur de Δ vaut donc : $\Delta = (12 + 8) - (3 + 3) = 14$.

On ajoute au graphe un sommet α , un sommet β , des arcs joignant le sommet α aux sommets excédentaires a et b , des arcs joignant les sommets déficitaires f , g et h au sommet β ; on considère alors le réseau \mathcal{R} correspondant à ce nouveau graphe, ayant α pour racine, β pour puits. On affecte aux arcs initiaux une capacité égale à la capacité du camion correspondant ; on affecte à un arc joignant α à un sommet excédentaire une capacité égale à

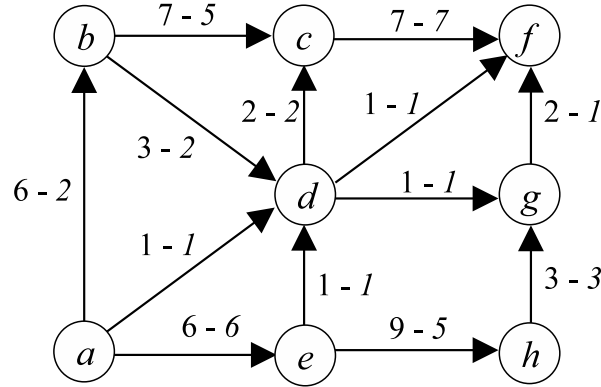


FIGURE 7.11 – Le graphe \mathcal{G} .

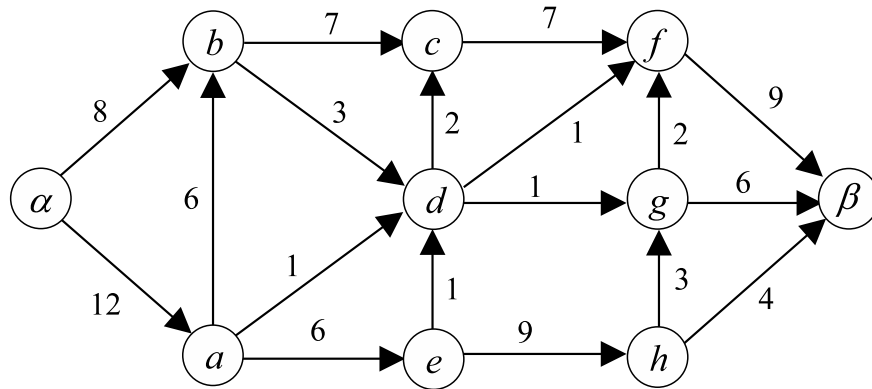


FIGURE 7.12 – Le réseau \mathcal{R} .

la valeur de l'excédent et à un arc joignant un sommet déficitaire au sommet β une capacité égale à la valeur du déficit. Le réseau \mathcal{R} est tracé sur la figure 7.12.

Clairement, un plan de transport correspond bijectivement à un flot dans \mathcal{R} ; à un plan de transport conduisant à une valeur Δ correspond un flot de valeur Δ ; il suffit donc de déterminer dans \mathcal{R} un flot de valeur maximum pour déterminer un plan de transport maximisant Δ . Le problème se ramène donc à un problème de flot maximum dans le graphe obtenu.

On utilise l'algorithme de Ford et Fulkerson. On initialise les flux avec les valeurs correspondant au plan de transport proposé (voir la figure 7.13 où les flux sont en italique).

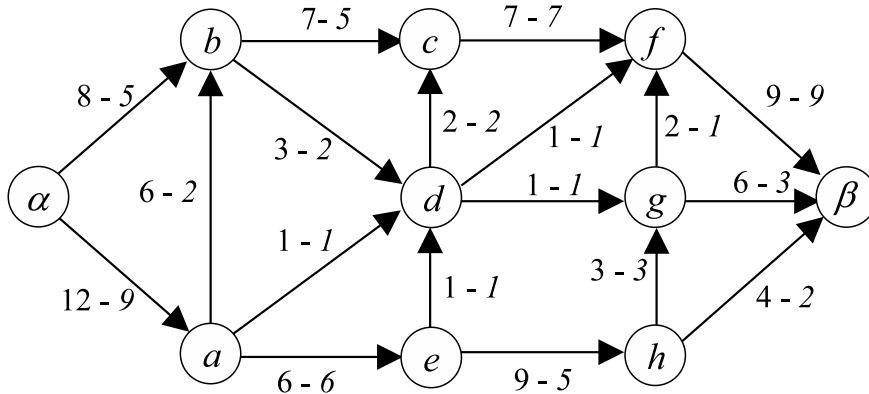


FIGURE 7.13 – Flot initial dans \mathcal{R} .

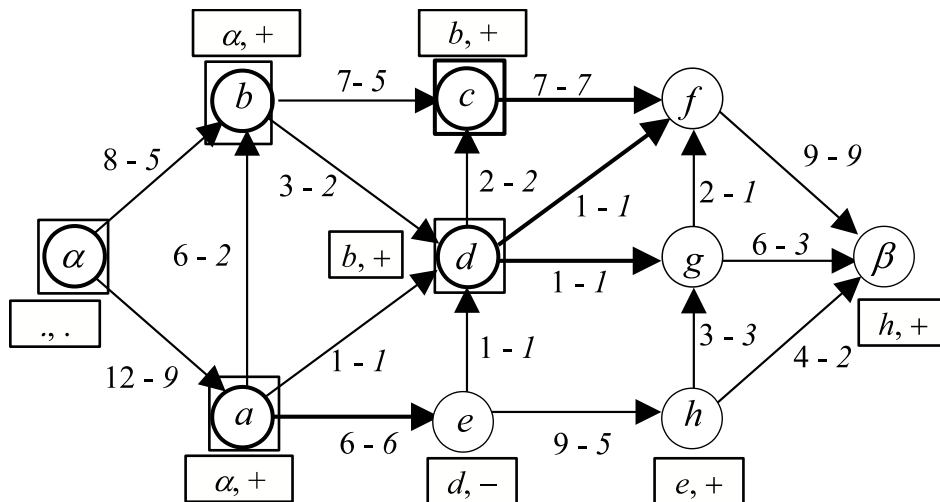


FIGURE 7.14 – Marquage dans \mathcal{R} .

On recherche une chaîne augmentante; on indique sur la figure 7.14 les marquages des sommets. Les sommets marqués sont en gras, les sommets examinés sont entourés d'un carré. On a ainsi trouvé la chaîne augmentante $\alpha \rightarrow b \rightarrow d \leftarrow e \rightarrow h \rightarrow \alpha$; cette chaîne est augmentant de 1. On augmente la valeur du flot en utilisant cette chaîne (voir la figure 7.15).

On cherche à nouveau une chaîne augmentante, le marquage correspondant est indiqué sur la figure 7.16. On n'a pas trouvé de chaîne augmentante,

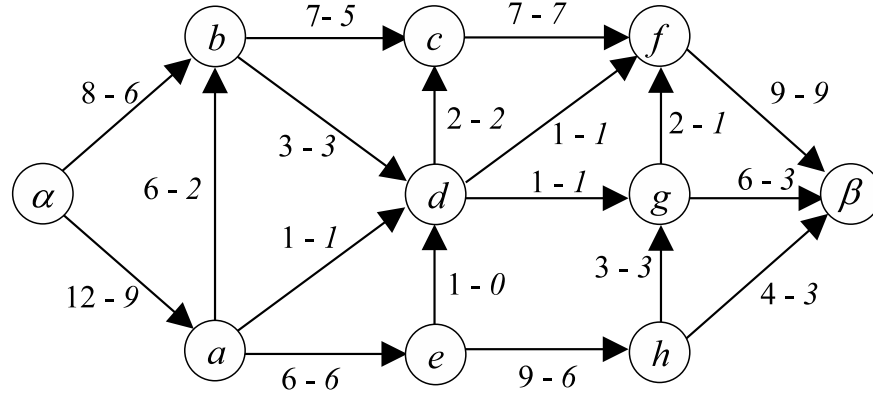


FIGURE 7.15 – Actualisation du flot dans \mathcal{R} .

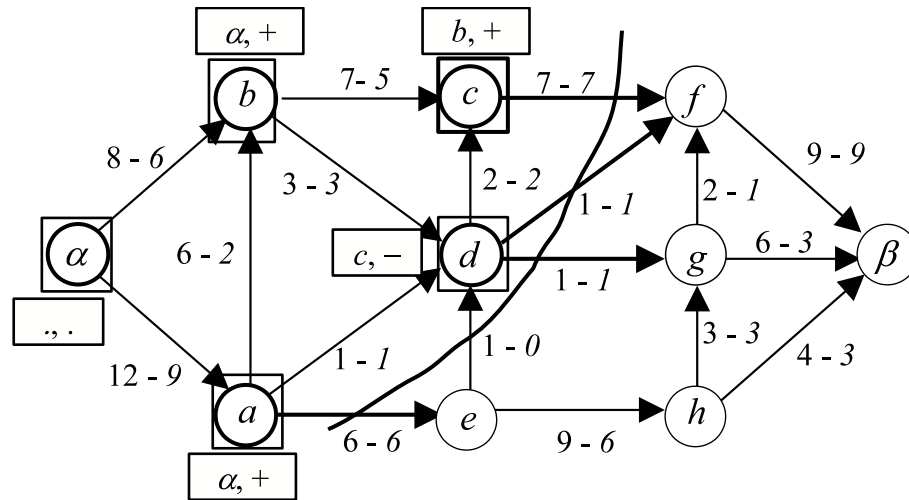


FIGURE 7.16 – Nouveau marquage dans \mathcal{R} .

le flot est donc de valeur maximum. Le maximum de Δ vaut 15 et les quantités à transporter par les camions sont égales aux flux sur les arcs correspondants.

Prouvons que la solution obtenue est optimale. Considérons la coupe où $M = \{s, a, b, c, d\}$; la capacité de cette coupe composée des arcs en gras sur la figure 7.16 vaut : $6 + 7 + 1 + 1 = 15$. On a une coupe de capacité 15 et un flot de même valeur 15; cela prouve que la coupe est de capacité minimum et le flot de valeur maximum.

7.7 Exercices

Exercice 7.1.

On considère le réseau tracé sur la figure 7.17. Le nombre encadré à côté d'un arc indique la capacité de l'arc ; l'autre nombre indique une valeur de flux sur l'arc. Déterminer un flot de valeur maximum de α à β dans ce réseau en utilisant les flux proposés pour initialiser l'algorithme. Exhiber une coupe de capacité minimum et vérifier ainsi que le flot calculé est bien maximum.

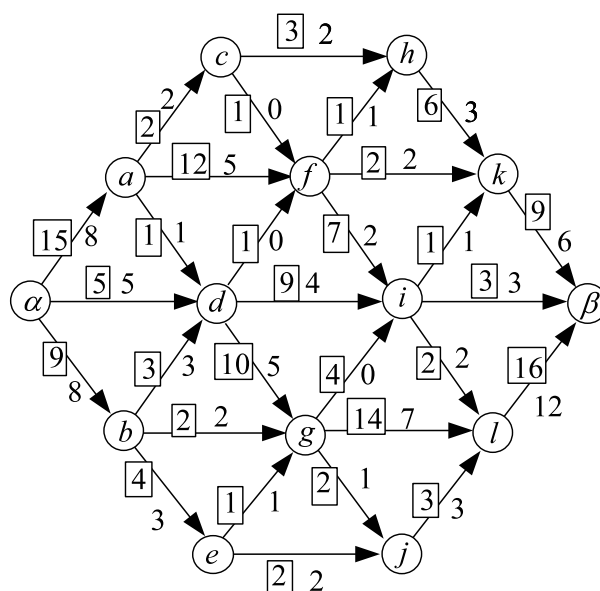


FIGURE 7.17 – Réseau pour l'exercice 1.

Corrigé. On commence par vérifier que les flux indiqués correspondent bien à un flot : les flux sont positifs ou nuls, ne dépassent pas les capacités et il y a conservation du flot en tout sommet autre que α et β . Il s'agit d'un flot de valeur 21. On applique l'algorithme de Ford et Fulkerson en utilisant l'algorithme de marquage. Le déroulement de celui-ci est résumé par le tableau ci-dessous.

sommets examinés	sommets marqués et marques
α	$a(\alpha, +), b(\alpha, +)$
a	$f(a, +)$
b	$e(b, +)$
f	$i(f, +)$
e	
i	$d(i, -)$
d	$g(d, +)$
g	$j(g, +), l(g, +)$
j	
l	$\beta(l, +)$

Le puits a été marqué : il existe une chaîne augmentante. Pour reconstituer celle-ci, on utilise les marques, en commençant par β et en remontant la chaîne à l'envers ; on obtient ainsi la chaîne $\alpha \rightarrow a \rightarrow f \rightarrow i \leftarrow d \rightarrow g \rightarrow l \rightarrow \beta$. Cette chaîne est augmentante de 4. On actualise le flot ; le nouveau flot est représenté sur la figure 7.18.

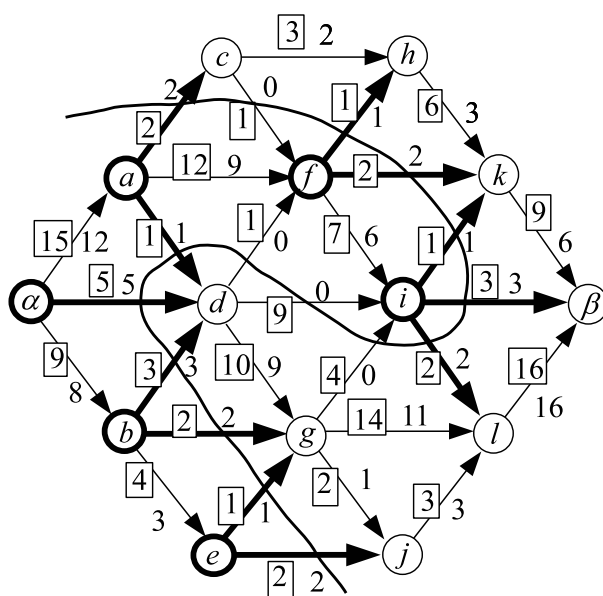


FIGURE 7.18 – Flot actualisé pour l'exercice 7.1..

On applique de nouveau l'algorithme de marquage, à partir du nouveau flot. On obtient le tableau ci-dessous.

sommets examinés	sommets marqués et marques
α	$a(\alpha, +), b(\alpha, +)$
a	$f(a, +)$
b	$e(b, +)$
f	$i(f, +)$
e	
i	

Le puits n'a pas été marqué; le flot actuel, de valeur 25, est de valeur maximum. L'ensemble des sommets marqués est $M = \{\alpha, a, b, e, f, i\}$. La coupe (M, \overline{M}) est de capacité minimum, égale à la valeur du flot, c'est-à-dire 25. Les sommets marqués dans cette dernière étape ainsi que les arcs de la coupe (M, \overline{M}) sont en gras sur la figure 7.18. On peut vérifier que les arcs de (M, \overline{M}) sont saturés alors que ceux de (\overline{M}, M) portent un flux nul, ce qui confirme l'exactitude du résultat.

Exercice 7.2. Le but de cet exercice est de montrer que l'algorithme de Ford et Fulkerson peut échouer avec des capacités irrationnelles³.

On considère le réseau \mathcal{R} de la figure 7.19. La capacité r de l'arc (d, c) est donnée par : $r = \frac{\sqrt{5} - 1}{2} \simeq 0,618$.

On définit les trois chaînes suivantes :

$$C_1 = \alpha \rightarrow d \rightarrow c \leftarrow b \rightarrow a \rightarrow \beta$$

$$C_2 = \alpha \rightarrow b \rightarrow c \leftarrow d \rightarrow \beta$$

$$C_3 = \alpha \rightarrow a \leftarrow b \rightarrow c \rightarrow \beta.$$

Un flot f étant défini, on appelle *capacité résiduelle (par rapport à f)* d'un arc a la différence $cr_f(a) = c(a) - f(a)$.

On part du flot nul dans \mathcal{R} . On applique l'algorithme de Ford et Fulkerson en utilisant d'abord la chaîne augmentante $C_0 = \alpha \rightarrow b \rightarrow c \rightarrow \beta$ puis, cycliquement, les chaînes C_1, C_2, C_1, C_3 . Pour alléger les calculs, on admet que, pour les questions 1 et 2, les arcs issus de α ainsi que les arcs d'extrémité β ne sont jamais saturés.

3. L'exemple étudié ici est dû à U. Zwick (The smallest networks on which the Ford-Fulkerson maximum flow procedure may fail to terminate, *Theoretical Computer Science* 148 (1), 1995, 165-170). U. Zwick donne deux autres exemples, dont un à 6 sommets et 8 arcs, précisant qu'il s'agit d'un plus petit exemple, à la fois en nombre de sommets et en nombre d'arcs. L.R. Ford et D.R. Fulkerson avaient quant à eux exhibé un exemple à 10 sommets et 48 arcs.

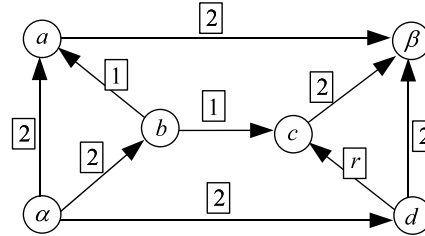


FIGURE 7.19 – Réseau pour l’exercice 7.2..

Q1. Partir du flot nul et appliquer successivement les chaînes augmentantes $C_0, C_1, C_2, C_1, C_3, C_1, C_2, C_1, C_3$. Calculer les capacités résiduelles des arcs (b, a) , (d, c) et (b, c) après l’application de chacune de ces chaînes ; on indiquera aussi à chaque étape la quantité Δf dont augmente la valeur du flot. On pourra utiliser la formule : $r^2 = 1 - r$ ($\simeq 0,382$).

Q2. On continue l’application de l’algorithme de Ford et Fulkerson en utilisant cycliquement les chaînes C_1, C_2, C_1, C_3 . La suite des valeurs de flot obtenues converge-t-elle et, si oui, vers quelle valeur ?

Q3. Déterminer un flot de valeur maximum dans \mathcal{R} avec l’algorithme de Ford et Fulkerson à partir du flot nul et en appliquant la variante d’Edmonds et Karp : on mettra les sommets marqués dans une file et on examinera les sommets dans l’ordre induit par cette file (autrement dit, on examine le sommet qui a été marqué le premier parmi les sommets marqués non encore examinés). Pour marquer les sommets, on privilégiera l’ordre alphabétique.

Q4. On modifie les capacités des arcs d’origine α ou d’extrémité β , actuellement égales à 2, en leur donnant une valeur K avec $K \geq 2$. Cela change-t-il le déroulement effectué dans les questions 1 et 2 ? Quelle est maintenant la valeur du flot maximum ? Que peut-on en déduire pour l’algorithme de Ford et Fulkerson ?

Corrigé.

Q1. Les résultats demandés sont indiqués dans le tableau ci-dessous :

7.7. EXERCICES

123

étape	chaîne	Δf	$cr_f(b, a)$	$cr_f(d, c)$	$cr_f(b, c)$
0			1	r	1
1	C_0	1	$1 (=r^0)$	r	0
2	C_1	r	r^2	0	r
3	C_2	r	r^2	r	0
4	C_1	r^2	0	r^3	r^2
5	C_3	r^2	r^2	r^3	0
6	C_1	r^3	r^4	0	r^3
7	C_2	r^3	r^4	r^3	0
8	C_1	r^4	0	r^5	r^4
9	C_3	r^4	r^4	r^5	0

Q2. Une récurrence immédiate permet de voir que les résultats pour les étapes $4(p+1)+2$, $4(p+1)+3$, $4(p+1)+4$, $4(p+1)+5$ ($p \geq 0$) se déduisent de ceux des étapes $4p+2$, $4p+3$, $4p+4$, $4p+5$ par une multiplication par r^2 . Soit j compris entre 0 et 3 ; à l'issue de l'étape $4p+1+j$, la valeur du

flot vaut : $1 + 2 \sum_{i=1}^{2p} r^i + v(p, j)$ avec $v(p, j) = 0$ si $j = 0$, $v(p, j) = r^{2p+1}$ si $j = 1$, $v(p, j) = 2r^{2p+1}$ si $j = 2$, $v(p, j) = 2r^{2p+1} + r^{2p+2}$ si $j = 3$. Quand p tend vers l'infini, $v(p, j)$ tend vers 0 et la valeur du flot converge donc vers :

$$1 + 2 \sum_{i=1}^{\infty} r^i = 1 + 2 \frac{r}{1-r} = 1 + 2 \frac{1-r^2}{1-r} = 3 + 2r \simeq 4,236.$$

Q3. Le déroulement de l'algorithme est illustré par la figure 7.20.

- Première étape : on marque α , on examine α , on marque a , b et d , on examine a , on marque β et on s'arrête ; on a trouvé la chaîne augmentante $\alpha \rightarrow a \rightarrow \beta$ qui permet d'augmenter de 2 la valeur du flot.
- Deuxième étape : on marque α , on examine α , on marque b et d , on examine b , on marque a et c , on examine d , on marque β : on a trouvé la chaîne augmentante $\alpha \rightarrow d \rightarrow \beta$ qui permet d'augmenter de 2 la valeur du flot.
- Troisième étape : on marque α , on examine α , on marque b , on examine b , on marque a et c , on examine a (ce qui ne donne rien), on examine c , on marque β : on a trouvé la chaîne augmentante $\alpha \rightarrow b \rightarrow c \rightarrow \beta$ qui permet d'augmenter de 1 la valeur du flot.
- Quatrième étape : on marque α , on examine α , on ne marque aucun sommet, l'algorithme est terminé.
- Le flot est de valeur maximum, valeur qui est égale à 5 (on peut remarquer

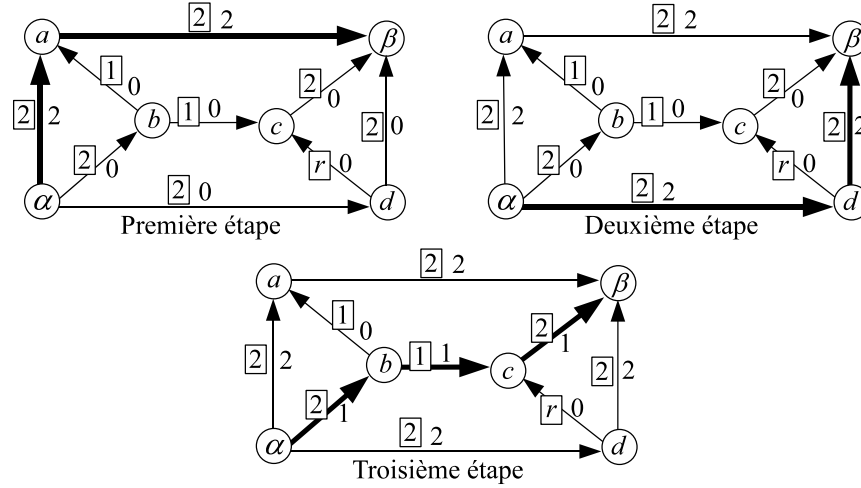


FIGURE 7.20 – Déroulement de l’algorithme pour la question 3 de l’exercice 7.2..

que la coupe qui isole α , seul sommet marqué lors de la dernière étape, du reste des sommets est bien de capacité 5).

Q4. Le déroulement effectué dans les questions 1 et 2 ne change pas et la valeur du flot converge toujours vers $3 + 2r$, alors qu’il est facile d’établir que la valeur maximum d’un flot est maintenant $2K + 1$. On déduit de ce qui précède que, si les chaînes augmentantes sont mal choisies, l’algorithme de Ford et Fulkerson peut échouer à déterminer un flot de valeur maximum quand les capacités ne sont pas des nombres entiers (ou, de manière équivalente, rationnels). En choisissant K assez grand, on peut même faire en sorte que l’algorithme de Ford et Fulkerson ne converge que vers une valeur arbitrairement éloignée de la valeur optimale. En revanche, la variante d’Edmonds et Karp permet bien de déterminer un flot de valeur maximum à l’aide d’un nombre fini d’itérations, que les capacités soient entières (ou rationnelles) ou non.

Exercice 7.3. On considère le problème (P) du flot maximum dans un réseau \mathcal{R} utilisant un graphe $\mathcal{G} = (S, A)$, de source α , de puits β . Quitte à l’ajouter si cet arc n’y est pas déjà, on suppose que A contient l’arc (β, α) avec une capacité suffisamment grande pour ne pas limiter la valeur de la fonction objectif dans les problèmes (P') et (P'') définis ci-dessous (on peut la prendre

7.7. EXERCICES

125

par exemple égale à la somme des autres capacités du réseau). Néanmoins, on ne considérera pour le problème (P) que les flots f avec $f(\beta, \alpha) = 0$, car cette égalité est vérifiée à l’optimum de (P) .

Q1. Montrer que le problème (P) est équivalent au problème (P') suivant :

$$\begin{aligned} & \text{Maximiser } f_{\beta, \alpha} \\ & \text{avec les contraintes : } \begin{cases} \forall s \in S, \sum_{\{t|(t,s) \in A\}} f_{t,s} - \sum_{\{t|(s,t) \in A\}} f_{s,t} = 0 \\ \forall (s, t) \in A, 0 \leq f_{s,t} \leq c(s, t). \end{cases} \end{aligned}$$

Q2. Montrer que le problème (P') peut s’écrire sous la forme (P'') suivante (forme standard), avec les mêmes variables que ci-dessus :

$$\begin{aligned} & \text{Maximiser } f_{\beta, \alpha} \\ & \text{avec les contraintes : } \begin{cases} \forall s \in S, \sum_{t|(t,s) \in A} f_{t,s} - \sum_{t|(s,t) \in A} f_{s,t} \leq 0 \\ \forall (s, t) \in A, 0 \leq f_{s,t} \leq c(s, t). \end{cases} \end{aligned}$$

Q3. Écrire le problème dual (D) du problème (P'') . On notera y_s une variable duale correspondant à un sommet s et $y_{s,t}$ une variable duale correspondant à un arc (s, t) .

Q4. Soit (M, \overline{M}) une coupe dans le réseau \mathcal{R} . On pose pour tout sommet $s \in S$:

$$\begin{cases} y_s = 1 \text{ si } s \in M \\ y_s = 0 \text{ sinon.} \end{cases}$$

On pose pour tout arc $(s, t) \in A$:

$$\begin{cases} y_{s,t} = 1 \text{ si } (s, t) \in (M, \overline{M}) \\ y_{s,t} = 0 \text{ sinon.} \end{cases}$$

Montrer que l’ensemble des valeurs ainsi définies donne une solution réalisable du problème (D) . On appelle celle-ci « solution associée à la coupe (M, \overline{M}) ». Indiquer une relation entre la capacité de la coupe (M, \overline{M}) et la valeur de la fonction duale pour la solution associée à cette coupe.

Q5. Soit (M, \overline{M}) une coupe de capacité minimum dans le réseau \mathcal{R} . Montrer que la solution de (D) associée à cette coupe est une solution optimale de (D) .

Corrigé.

Q1. Considérons un flot f dans \mathcal{R} , de valeur $val(f)$ et avec $f(\beta, \alpha) = 0$. On

pose : $f_{\beta,\alpha} = \text{val}(f)$ et, pour $(s,t) \neq (\beta,\alpha)$, $f_{s,t} = f(s,t)$. Les contraintes de (P') sont satisfaites et on a ainsi une solution réalisable de (P') de valeur $\text{val}(f)$.

Réciproquement, on considère une solution réalisable du problème (P') . On pose $f(\beta,\alpha) = 0$ et, pour $(s,t) \neq (\beta,\alpha)$, $f(s,t) = f_{s,t}$. On obtient ainsi un flot dans \mathcal{R} , de valeur $f_{\beta,\alpha}$.

Les deux problèmes (P) et (P') ont donc mêmes solutions réalisables, sauf pour le couple (β,α) et les valeurs prises par leurs fonctions objectifs sont égales : les deux problèmes (P) et (P') sont équivalents.

Q2. Il est clair qu'une solution réalisable du problème (P') est une solution réalisable pour le problème (P'') dont les contraintes sont apparemment moins fortes.

Considérons une solution réalisable pour le problème (P'') . On a :

$$\sum_{(s,t) \in A} f_{s,t} = \sum_{s \in S} \sum_{t|(s,t) \in A} f_{s,t} \text{ et } \sum_{(s,t) \in A} f_{s,t} = \sum_{t \in S} \sum_{s|(s,t) \in A} f_{s,t} = \sum_{s \in S} \sum_{t|(t,s) \in A} f_{t,s}.$$

D'où :

$$\sum_{s \in S} \left[\sum_{t|(t,s) \in A} f_{t,s} - \sum_{t|(s,t) \in A} f_{s,t} \right] = 0.$$

Supposons qu'il existe un sommet s avec $\sum_{t|(t,s) \in A} f_{t,s} - \sum_{t|(s,t) \in A} f_{s,t} < 0$.

Alors il existe un sommet s' vérifiant $\sum_{t|(t,s') \in A} f_{t,s'} - \sum_{t|(s',t) \in A} f_{s',t} > 0$, ce qui

contredit la contrainte du problème (P'') en le sommet s' . On a donc bien, pour tout sommet s , $\sum_{t|(t,s) \in A} f_{t,s} - \sum_{t|(s,t) \in A} f_{s,t} = 0$. La solution réalisable du

problème (P'') est donc aussi une solution réalisable pour le problème (P') .

D'où l'équivalence entre les deux problèmes (P') et (P'') .

Q3. Pour définir le problème dual, on introduit les variables comme indiqué. Pour $s \in S$, la variable y_s correspond à la contrainte :

$$\sum_{t|(t,s) \in A} f_{t,s} - \sum_{t|(s,t) \in A} f_{s,t} \leq 0.$$

7.7. EXERCICES

127

Pour $(s, t) \in A$, la variable $y_{s,t}$ correspond à la contrainte : $f_{s,t} \leq c(s, t)$. Le problème dual (D) est donc :

$$\begin{aligned} \text{Minimiser } w &= \sum_{(s,t) \in A} c(s,t)y_{s,t} \\ \text{avec les contraintes : } &\begin{cases} y_\alpha - y_\beta + y_{\beta,\alpha} \geq 1 \\ \forall (s,t) \in A \text{ avec } (s,t) \neq (\beta, \alpha), y_t - y_s + y_{s,t} \geq 0 \\ \forall (s,t) \in A, y_{s,t} \geq 0 \\ \forall s \in S, y_s \geq 0. \end{cases} \end{aligned}$$

Q4. Plusieurs cas se présentent :

- si $(s, t) \in (M, \overline{M})$, alors $y_{s,t} = y_s = 1, y_t = 0$, d'où $y_t - y_s + y_{s,t} = 0$;
- si $(s, t) \in A \setminus (M, \overline{M})$ avec $(s, t) \neq (\beta, \alpha)$, alors $y_{s,t} = 0$; si s et t sont du même côté de la coupe, on a $y_s = y_t$ et, sinon, $(s, t) \in (\overline{M}, M)$ d'où $y_s = 0$ et $y_t = 1$; dans tous ces cas, on a $y_t - y_s + y_{s,t} \geq 0$;
- si $(s, t) = (\beta, \alpha)$, on a $y_\alpha = 1, y_\beta = 0$ et $y_{\beta,\alpha} = 0$; d'où $y_\alpha - y_\beta + y_{\beta,\alpha} \geq 1$.

Toutes les contraintes de (D) sont vérifiées. De plus, on a pour cette solution : $w = \sum_{(s,t) \in (M, \overline{M})} c(s,t)$. La valeur de w est égale à la capacité de (M, \overline{M}) .

Q5. Soit (M, \overline{M}) une coupe de capacité minimum dans \mathcal{R} . D'après le théorème 7.10., la capacité de (M, \overline{M}) est égale à la valeur maximum d'un flot. Or, (D) est le problème dual de la recherche d'un flot de valeur maximum. Donc, d'après les résultats de la dualité linéaire, le minimum de la fonction duale w est égal à la valeur maximum d'un flot, c'est-à-dire aussi à la capacité de (M, \overline{M}) . En utilisant la question précédente, on obtient que la solution de (D) associée à (M, \overline{M}) est une solution réalisable qui atteint le minimum de (D) , c'est donc une solution optimale de (D) .

En ce sens, on peut considérer que le problème dual de la recherche d'un flot de valeur maximum est la recherche d'une coupe de capacité minimum.

Exercice 7.4. Une entreprise dispose de quatre machines m_1, m_2, m_3 et m_4 et souhaite effectuer sept tâches $t_1, t_2, t_3, t_4, t_5, t_6$ et t_7 . Les machines m_1 et m_3 ne peuvent effectuer qu'une seule tâche, la machine m_2 peut effectuer trois tâches et la machine m_4 peut effectuer deux tâches. Les tâches

ne nécessitent qu’une seule machine pour être exécutées mais ne peuvent pas l’être par n’importe quelle machine ; le tableau ci-dessous indique pour chaque tâche les machines capables de les exécuter.

	m_1	m_2	m_3	m_4
t_1		★		★
t_2	★		★	
t_3		★		★
t_4	★		★	
t_5		★	★	
t_6	★		★	
t_7		★		★

Il s’agit d’affecter les tâches aux machines de façon à maximiser le nombre de tâches effectuées.

Q1. Modéliser ce problème d’affectation de tâches.

Q2. Résoudre le problème à l’aide de l’algorithme de Ford et Fulkerson en partant d’un flot correspondant à la situation suivante :

- la tâche t_1 est exécutée par la machine m_2
- la tâche t_3 est exécutée par la machine m_4
- la tâche t_4 est exécutée par la machine m_1
- la tâche t_5 est exécutée par la machine m_3
- la tâche t_7 est exécutée par la machine m_2 .

Q3. Prouver par un argument théorique que le flot obtenu à la question précédente est bien de valeur maximum.

Corrigé.

Q1. On modélise le problème à l’aide du réseau \mathcal{R} de la figure 7.21.a. ; sur cette figure, les capacités non indiquées valent 1 et les nombres encadrés sont les capacités différentes de 1. Par construction, il y a une bijection entre les façons d’affecter les tâches aux machines et les flots à flux entiers dans \mathcal{R} ; de plus, la valeur du flot est égal au nombre de tâches effectuées. Déterminer comment affecter les tâches aux machines de façon à maximiser le nombre de tâches effectuées revient à chercher un flot à flux entiers de valeur maximum dans \mathcal{R} .

7.7. EXERCICES

129

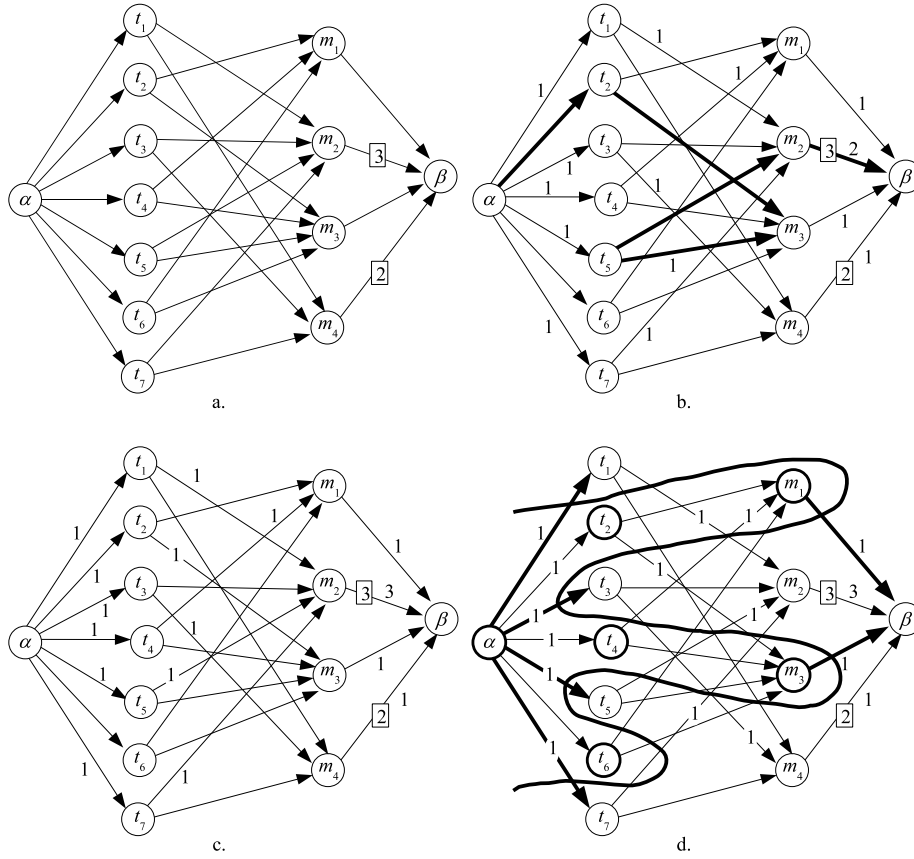


FIGURE 7.21 – Réseau et déroulement pour l’exercice 7.4.. Les flux nuls ne sont pas mentionnés.

Q2. Les affectations servant de point de départ correspondent au flot représenté par la figure 7.21.b.; les flux nuls ne sont pas indiqués sur la figure. On applique une étape de l’algorithme de Ford et Fulkerson résumée par le tableau ci-dessous :

sommets examinés	sommets marqués et marques
α	$t_2 (\alpha, +), t_6 (\alpha, +)$
t_2	$m_1 (t_2, +), m_3 (t_2, +)$
t_6	
m_1	$t_4 (m_1, -)$
m_3	$t_5 (m_3, -)$
t_4	
t_5	$m_2 (t_5, +)$
m_2	$\beta (m_2, +), t_1 (m_2, -)$

Le puits, β , a été marqué. On a trouvé la chaîne augmentante $\alpha \rightarrow t_2 \rightarrow m_3 \leftarrow t_5 \rightarrow m_2 \rightarrow \beta$. Cette chaîne et le nouveau flot sont représentés respectivement sur les figures 7.21.b. et 7.21.c. On effectue une nouvelle étape de l’algorithme de Ford et Fulkerson résumée par le tableau ci-dessous :

sommets examinés	sommets marqués et marques
α	$t_6 (\alpha, +)$
t_6	$m_1 (t_6, +), m_3 (t_6, +)$
m_1	$t_4 (m_1, -)$
m_3	$t_2 (m_3, -)$
t_4	
t_2	

Le puits n’a pas été marqué, le flot actuel est optimum. On peut exécuter six tâches en plaçant la tâche t_1 sur la machine m_2 , la tâche t_2 sur la machine m_3 , la tâche t_3 sur la machine m_4 , la tâche t_4 sur la machine m_1 , la tâche t_5 sur la machine m_2 , la tâche t_7 sur la machine m_2 .

Q3. Considérons la coupe (M, \overline{M}) , où $M = \{\alpha, t_2, t_4, t_6, m_1, m_3\}$ est l’ensemble des sommets marqués à la dernière étape (voir la figure 7.21.d.). Elle est de capacité 6, comme la valeur du flot. Le flot est donc de valeur maximum et la coupe de capacité minimum.

Exercice 7.5. On dispose de deux processeurs P_1 et P_2 pour exécuter n tâches nommées T_1, T_2, \dots, T_n . Pour chaque tâche T_k ($1 \leq k \leq n$), on connaît le coût $t_{k,i}$ d’exécution de T_k par le processeur P_i ($1 \leq i \leq 2$). Les deux processeurs n’ayant pas les mêmes performances, on n’a pas nécessairement $t_{k,1} = t_{k,2}$.

7.7. EXERCICES

131

Par ailleurs, il est parfois nécessaire d'établir une communication entre une tâche T_i s'exécutant sur P_1 et une autre tâche T_j s'exécutant sur P_2 . Un coût de communication non nul $\tau_{i,j} = \tau_{j,i}$ est alors à payer ; pour les tâches T_i et T_j ne communiquant pas entre elles, $\tau_{i,j}$ est nul.

On cherche, en déterminant judicieusement pour chaque tâche le processeur qui l'exécutera, à minimiser le coût total d'exécution de toutes les tâches et de communication entre elles. On note K_1 (respectivement K_2) l'ensemble des tâches exécutées par P_1 (respectivement P_2) et θ_{K_1,K_2} le coût total lié à ce placement des tâches. On a plus précisément :

$$\theta_{K_1,K_2} = \sum_{T_k \in K_1} t_{k,1} + \sum_{T_k \in K_2} t_{k,2} + \sum_{(T_i,T_j) \in K_1 \times K_2} \tau_{i,j}.$$

On cherche à placer les tâches de façon à minimiser θ_{K_1,K_2} . Un tel placement est dit optimal.

Exemple numérique avec $n = 4$.

Les coûts d'exécution sont :

$$t_{1,1} = 1 ; t_{2,1} = 3 ; t_{3,1} = 5 ; t_{4,1} = 1 ; t_{1,2} = 2 ; t_{2,2} = 6 ; t_{3,2} = 3 ; t_{4,2} = 1.$$

Les coûts de communication non nuls sont :

$$\tau_{1,2} = \tau_{2,1} = 3 ; \tau_{2,3} = \tau_{3,2} = 1 ; \tau_{3,4} = \tau_{4,3} = 2.$$

Ainsi, pour $K_1 = \{T_1, T_4\}$ et $K_2 = \{T_2, T_3\}$, on a :

$$\theta_{K_1,K_2} = (1 + 1) + (6 + 3) + (3 + 2) = 16.$$

Q1. On veut modéliser le problème sous la forme de la recherche d'une coupe de capacité minimum. Déterminer, dans le cas général, un réseau dont les sommets sont P_1 (la source), P_2 (le puits) et les sommets T_1, \dots, T_n , en choisissant les arcs et les capacités de sorte que θ_{K_1,K_2} représente la capacité d'une coupe séparant P_1 de P_2 . Préciser cette coupe. En déduire une méthode pour déterminer un placement optimal.

Indication : un arc entre P_1 et un sommet T_k aura pour capacité $t_{k,1}$.

Q2. Tracer le réseau correspondant à l'exemple numérique donné ci-dessus ; on précisera en particulier les capacités des arcs. Déterminer un placement optimal pour cet exemple en utilisant la méthode établie à la question 1. On précisera l'algorithme permettant de mettre en œuvre la méthode ; on partira du flot nul puis on exploitera successivement les cinq chaînes augmentantes suivantes :

$$P_1 \rightarrow T_1 \rightarrow P_2, \quad P_1 \rightarrow T_1 \rightarrow T_2 \rightarrow P_2, \quad P_1 \rightarrow T_2 \rightarrow P_2, \quad P_1 \rightarrow T_3 \rightarrow P_2, \quad P_1 \rightarrow T_4 \rightarrow P_2.$$

On prouvera que la solution est optimum en utilisant un argument de nature théorique et ne s'appuyant pas sur le déroulement de l'algorithme.

Corrigé.

Q1. On considère le réseau \mathcal{R} dont la source, le puits et l'ensemble des sommets sont décrits dans l'énoncé. Pour $1 \leq k \leq n$, on ajoute un arc de P_1 à T_k avec la capacité $t_{k,2}$ et un arc de T_k à P_2 avec la capacité $t_{k,1}$. Enfin, pour tout couple (i, j) ($1 \leq i \leq n$, $1 \leq j \leq n$) avec $\tau_{i,j} \neq 0$, on ajoute un arc de T_i à T_j et un arc de T_j à T_i avec la capacité $\tau_{i,j}$.

Soient $K_1 \subseteq \{T_1, T_2, \dots, T_n\}$ et $K_2 = \{T_1, T_2, \dots, T_n\} \setminus K_1$. On considère la coupe (S, \bar{S}) avec $S = \{P_1\} \cup K_1$. En s'appuyant sur la figure 7.22, on vérifie immédiatement l'égalité : $\text{capacité}(S, \bar{S}) = \theta_{K_1, K_2}$. Plus formellement, on a ainsi défini une bijection entre les placements sur les deux processeurs et les coupes dans \mathcal{R} , le coût d'un placement étant égal à la capacité de la coupe qui lui correspond.

Pour déterminer un placement optimal, il suffit donc de chercher une coupe de capacité minimum dans \mathcal{R} .

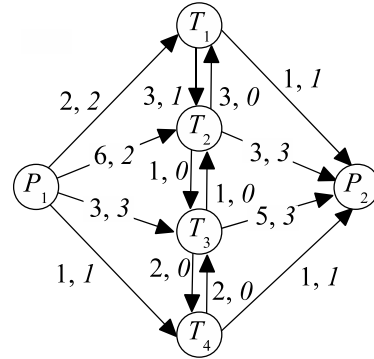
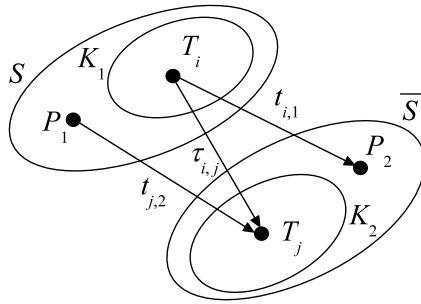


FIGURE 7.22 – Les arcs de (S, \bar{S}) . FIGURE 7.23 – Réseau et flux après utilisation des cinq chaînes.

Q2. Le réseau est tracé sur la figure 7.23. Sur les arcs, les capacités sont indiquées en caractères romains, les flux obtenus après utilisation des cinq chaînes sont en italiques.

On applique l'algorithme de Ford et Fulkerson. En partant du flot nul :

- la chaîne $P_1 \rightarrow T_1 \rightarrow P_2$ permet d'augmenter la valeur du flot de 1 ;
- la chaîne $P_1 \rightarrow T_1 \rightarrow T_2 \rightarrow P_2$ permet d'augmenter la valeur du flot de 1 ;

7.7. EXERCICES

133

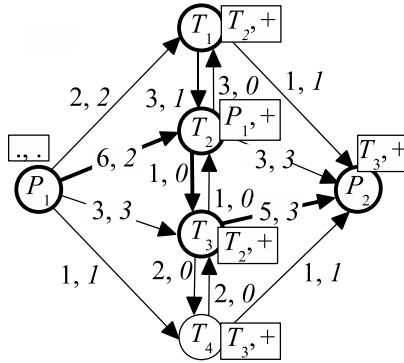


FIGURE 7.24 – Chaîne obtenue.

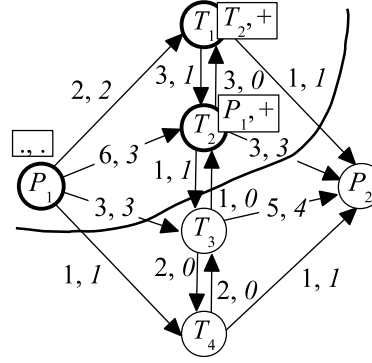


FIGURE 7.25 – Coupe minimum.

- la chaîne $P_1 \rightarrow T_2 \rightarrow P_2$ permet d'augmenter la valeur du flot de 2 ;
- la chaîne $P_1 \rightarrow T_3 \rightarrow P_2$ permet d'augmenter la valeur du flot de 3 ;
- la chaîne $P_1 \rightarrow T_4 \rightarrow P_2$ permet d'augmenter la valeur du flot de 1.

On cherche une chaîne augmentante. La figure 7.24 illustre cette recherche. Les marques sont inscrites à côté des sommets marqués. On marque P_1 . En examinant P_1 , on marque T_2 avec $(P_1, +)$. En examinant T_2 , on marque T_1 et T_3 avec $(T_2, +)$. En examinant T_1 , on ne marque aucun sommet. En examinant T_3 , on marque P_2 et T_4 avec $(T_3, +)$. Le sommet P_2 est marqué : on a déterminé la chaîne augmentante $P_1 \rightarrow T_2 \rightarrow T_3 \rightarrow P_2$, qui permet d'augmenter la valeur du flot de 1. Le nouveau flot est donné par la figure 7.25.

On cherche à nouveau une chaîne augmentante. On marque P_1 . En examinant P_1 , on marque T_2 avec $(P_1, +)$. En examinant T_2 , on marque T_1 avec $(T_2, +)$. En examinant T_1 , on ne marque aucun sommet. Tous les sommets marqués ont été examinés, le flot est de valeur maximum ; la valeur de ce flot vaut 9. Les sommets marqués lors de la dernière itération définissent la coupe (M, \overline{M}) avec $M = \{P_1, T_1, T_2\}$. La capacité de cette coupe (M, \overline{M}) vaut aussi 9, ce qui confirme que le flot est de valeur maximum et que la coupe (M, \overline{M}) est de capacité minimum.

Le placement optimal est obtenu en exécutant T_1 et T_2 sur P_1 , T_3 et T_4 sur P_2 . Le coût de l'opération est de 9.

Troisième partie

Optimisation continue non linéaire

Chapitre 8

Optimisation non linéaire sans contrainte

8.1 Introduction

Nous nous intéressons dans ce chapitre à la minimisation de fonctions définies sur \mathbb{R}^n et à valeurs dans \mathbb{R} . Les théorèmes et les méthodes seront donc décrits pour la minimisation. Le cas de la maximisation s'en déduit directement puisque maximiser une fonction f , c'est minimiser son opposée, grâce à la relation :

$$\text{maximum}_{x \in \mathbb{R}^n} f(x) = -\text{minimum}_{x \in \mathbb{R}^n} (-f)(x).$$

Soit f une fonction de \mathbb{R}^n dans \mathbb{R} .

Définition 8.1. On dit que f atteint un *minimum* (respectivement *maximum*) *global* en un point x^* de \mathbb{R}^n si, pour tout $x \in \mathbb{R}^n$, on a $f(x) \geq f(x^*)$ (respectivement $f(x) \leq f(x^*)$). On dit aussi que x^* est *solution optimale* ou *minimale* (respectivement *maximale*) du problème de minimisation (respectivement maximisation) de f sur \mathbb{R}^n .

Définition 8.2. On dit que f atteint un *minimum* (respectivement *maximum*) *local* en un point x^* de \mathbb{R}^n s’il existe une boule B de rayon non nul centrée en x^* telle que, pour tout $x \in B$, on ait $f(x) \geq f(x^*)$ (respectivement $f(x) \leq f(x^*)$).

Nous étudierons d’abord le cas $n = 1$, c’est-à-dire l’optimisation unidimensionnelle, en donnant quelques méthodes spécifiques d’optimisation. L’optimisation unidimensionnelle servira souvent d’outil pour l’optimisation multidimensionnelle.

Nous reviendrons ensuite au cas général pour établir quelques résultats théoriques, en particulier pour les cas des fonctions quadratiques et des fonctions convexes. Nous détaillerons quelques méthodes d’optimisation : les méthodes de descente, la méthode des gradients conjugués et la méthode de Newton.

8.2 Optimisation unidimensionnelle

On considère ici une application f de \mathbb{R} dans \mathbb{R} que l’on cherche à minimiser. Il arrive que l’on puisse déterminer analytiquement le minimum de f sur \mathbb{R} . Sinon, on peut envisager d’appliquer une des méthodes suivantes.

8.2.1 Méthode de Newton

On suppose f de classe C^2 . La méthode de Newton¹ consiste à construire une suite (x_k) à partir d’un réel x_0 de la façon suivante. En x_k , on approche f par :

$$q(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2}f''(x_k)(x - x_k)^2.$$

1. I. Newton, *Philosophiæ naturalis principia mathematica*, Londres, 1726. La méthode de Newton s’applique en général pour la recherche d’un zéro d’une fonction de classe C^1 . On cherche ici un zéro de la dérivée de f , la fonction f étant supposée de classe C^2 .

On remarque la relation suivante : $q'(x) = f'(x_k) + f''(x_k)(x - x_k)$.

Si on a $f''(x_k) > 0$ (cas où f est strictement convexe² autour de x_k), on pose :

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)},$$

qui est le point où q atteint son minimum : $q'(x_{k+1}) = 0$.

Si on a $f''(x_k) \leq 0$, la méthode échoue.

Si f est de classe C^3 et si x_0 est choisi assez proche d'un minimum local x^* vérifiant $f''(x^*) > 0$, alors la suite (x_k) converge de façon quadratique (voir la définition dans la partie 8.7.2, page 151) vers x^* . Nous démontrerons ce résultat dans le cas des fonctions de plusieurs variables réelles (partie 8.8, page 156).

8.2.2 Dichotomie pour une fonction dérivable

Définition 8.3. On dit qu'une fonction est *unimodale* s'il existe un réel x^* pour lequel la fonction est strictement décroissante sur $] - \infty, x^*]$ et strictement croissante sur $[x^*, +\infty[$.

Le point x^* est alors minimum global de f .

On suppose ici que f est unimodale et dérivable. Le point x^* est l'unique point où la dérivée de f s'annule. La première étape consiste en la recherche de x_{min} et x_{max} encadrant x^* , autrement dit tels qu'on ait les deux relations $f'(x_{min}) < 0$ et $f'(x_{max}) > 0$.

Après cette première étape, on pose : $x = \frac{1}{2}(x_{min} + x_{max})$; si on a $f'(x) > 0$, on remplace x_{max} par x , sinon on remplace x_{min} par x ; on répète l'opération jusqu'à ce qu'un critère d'arrêt à préciser soit atteint (l'exercice 1 du chapitre 9, page 191, donne une illustration de cette méthode).

2. Voir la partie 8.5, page 148, pour un rappel de la définition de la convexité.

La longueur de l'intervalle étant à chaque itération divisée par 2, on montre que la convergence est linéaire de taux 0,5 (voir la définition dans la partie 8.7.2, page 151).

Pour déterminer x_{min} et x_{max} , on peut procéder comme suit (on suppose dans ce qui suit que $f'(0)$ n'est pas nul ; dans le cas contraire, 0 est la solution du problème) :

- définir un pas de déplacement $h > 0$
- si $f'(0) < 0$, faire
 - ★ $x_{min} \leftarrow 0$
 - ★ tant que $f'(h) < 0$, faire
 - ▷ $x_{min} \leftarrow h$
 - ▷ $h \leftarrow 2h$
 - ★ $x_{max} \leftarrow h$
- sinon (on a alors $f'(0) > 0$), faire
 - ★ $h \leftarrow -h$
 - ★ $x_{max} \leftarrow 0$
 - ★ tant que $f'(h) > 0$, faire
 - ▷ $x_{max} \leftarrow h$
 - ▷ $h \leftarrow 2h$
 - ★ $x_{min} \leftarrow h$.

Remarque. Si f n'est pas unimodale, la dichotomie est néanmoins applicable si on connaît x_{min} et x_{max} avec $x_{min} < x_{max}$, $f'(x_{min}) < 0$ et $f'(x_{max}) > 0$. Elle converge alors vers un minimum local qui peut ne pas être global.

8.2.3 Interpolation quadratique

La méthode part du principe suivant : on choisit d’abord, à l’aide d’un algorithme préliminaire, x_1 , x_2 et x_3 vérifiant $x_1 < x_2 < x_3$ ainsi que les inégalités $f(x_2) \leq f(x_1)$ et $f(x_2) \leq f(x_3)$. On approche f par une fonction quadratique q ayant les mêmes valeurs que f en x_1, x_2 et x_3 :

$$q(x) = f(x_1) \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} + f(x_2) \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} + f(x_3) \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)}.$$

Le minimum de q est atteint sur $[x_1, x_3]$ en un point dont l’abscisse s’exprime facilement en fonction de $x_1, x_2, x_3, f(x_1), f(x_2)$ et $f(x_3)$; on note x_4 ce point. La mise à jour des points x_1, x_2 et x_3 se fait selon les règles suivantes :

- si $f(x_4) \leq f(x_2)$
 - ★ si $x_4 \leq x_2$, le nouveau triplet est (x_1, x_4, x_2)
sinon le nouveau triplet est (x_2, x_4, x_3)
- sinon
 - ★ si $x_4 \leq x_2$, le nouveau triplet est (x_4, x_2, x_3)
sinon le nouveau triplet est (x_1, x_2, x_4) .

On peut montrer que, si f est assez régulière, la convergence est superlinéaire d’ordre 1,3 (voir la définition dans la partie 8.7.2, page 151).

8.2.4 Dichotomie sans dérivation pour une fonction unimodale

On suppose ici que f est unimodale (voir la définition 8.3., page 139). Au départ, à l’aide d’un algorithme préliminaire, on

choisit a et b avec $a < b$ et tels que le minimum de f soit atteint entre a et b . On partage alors, à l’aide de points d , c et e , l’intervalle $[a, b]$ en quatre sous-intervalles égaux : $c = (a + b)/2$, $d = (a + c)/2$, $e = (c + b)/2$.

En comparant les valeurs prises par f en a , b , c , d et e , on peut éliminer deux des sous-intervalles définis par ces points et affirmer que le minimum de f est atteint dans l’union de deux sous-intervalles contigus $[a', c']$ et $[c', b']$. La figure 8.1 illustre un tel cas. On recommence alors avec l’intervalle $[a', b']$. À chaque étape, la longueur de l’intervalle est divisée par 2. La vitesse de convergence est linéaire (voir la définition dans la partie 8.7.2, page 151).

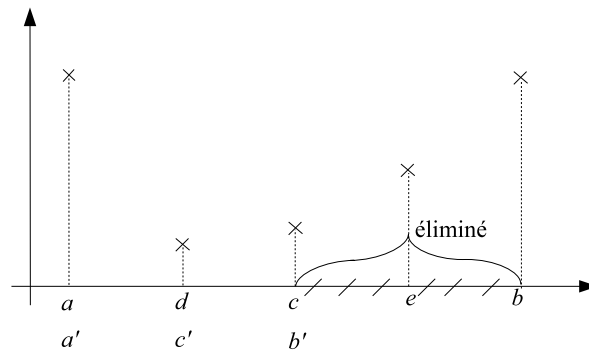


FIGURE 8.1 – Dichotomie sans dérivation.

8.3 Généralités pour l’optimisation multidimensionnelle

On considère ici des fonctions f définies sur \mathbb{R}^n et à valeurs dans \mathbb{R} . Dans la suite, quand on considérera une norme $\| \cdot \|$, il s’agira, sauf mention contraire, de la norme 2 (ou norme euclidienne pour un vecteur ; voir annexe ??, page ??) dans \mathbb{R}^n . On

8.3. GÉNÉRALITÉS POUR L’OPTIMISATION MULTIDIMENSIONNELLE¹⁴³

cherche à déterminer les points où f atteint des extrema locaux ou globaux. Pour cela, nous avons besoin de quelques définitions.

8.3.1 Notions de topologie

Nous donnons ci-dessous quelques notions de topologie³.

Définition 8.4. Une partie Ω de \mathbb{R}^n est un *ouvert* si, pour tout $x \in \Omega$, il existe une boule de rayon non nul et de centre x incluse dans Ω .

Définition 8.5. Une partie de \mathbb{R}^n est un *fermé* si son complémentaire est un ouvert.

L’ensemble \mathbb{R}^n et l’ensemble vide de \mathbb{R}^n sont à la fois ouverts et fermés. Tout produit de n intervalles ouverts de \mathbb{R} est un ouvert de \mathbb{R}^n ; de même, tout produit de n intervalles fermés de \mathbb{R} est un fermé de \mathbb{R}^n . Par exemple, l’ensemble $[0, 1] \times]-\infty, 3] \times \mathbb{R} \times [2, +\infty[$ est un fermé de \mathbb{R}^4 .

Définition 8.6. Une partie K de \mathbb{R}^n est un *compact* si elle est fermée et bornée.

Théorème 8.7. Une fonction f , à valeurs réelles, continue sur un ensemble compact non vide K de \mathbb{R}^n atteint ses bornes ; autrement dit, il existe $x_1 \in K$ et $x_2 \in K$ vérifiant, pour tout $x \in K$, $f(x_1) \leq f(x) \leq f(x_2)$: x_1 est un minimum global et x_2 un maximum global de f .

Dans tout ce chapitre, Ω désigne un ouvert de \mathbb{R}^n .

3. Voir par exemple H. Queffélec, *Topologie*, Dunod, 2016.

8.3.2 Gradient

Soit f une fonction d'un ouvert Ω de \mathbb{R}^n dans \mathbb{R} admettant en un point $x \in \Omega$ des dérivées partielles du premier ordre. On posera $x = (x_1, x_2, \dots, x_n)^t$ (les éléments de \mathbb{R}^n sont assimilés à des vecteurs-colonnes).

On note $\nabla f(x)$ et on appelle *gradient* de f au point x le vecteur-colonne :

$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}(x), \dots, \frac{\partial f}{\partial x_n}(x) \right)^t.$$

Si $F(x) = (f_1(x), \dots, f_p(x))$ est un vecteur-ligne où f_1, \dots, f_p sont des fonctions réelles de n variables réelles dérivables au point x , alors $\nabla F(x)$ est la matrice dont la j -ième colonne est $\nabla f_j(x)$.

Les formules suivantes nous seront utiles ultérieurement : si A est une matrice carrée constante d'ordre n , si $u(x)$ et $v(x)$ sont deux vecteurs-colonnes dépendant de x , alors :

$$\begin{aligned} \nabla (u^t A) &= \nabla (u^t) A \\ \nabla (u^t v) &= \nabla (u^t) v + \nabla (v^t) u. \end{aligned}$$

Si f admet en x^0 des dérivées partielles continues, on peut lui appliquer la *formule de Taylor* à l'ordre 1 :

$$f(x) = f(x^0) + (x - x^0)^t \nabla f(x^0) + \|x - x^0\| \varepsilon(x)$$

où $\varepsilon(x)$ tend vers 0 quand x tend vers x^0 .

Remarques.

1. Supposons f de classe C^1 . Si on considère la surface S de \mathbb{R}^{n+1} d'équation $x_{n+1} = f(x_1, \dots, x_n)$, alors l'expression $x_{n+1} = f(x^0) + (x - x^0)^t \nabla f(x^0)$ donne l'équation de l'hyperplan tangent à S au point $(x^0, f(x^0))$.
2. Nous nous intéresserons par la suite aux variations de f dans

8.4. CONDITION NÉCESSAIRE ET CONDITION SUFFISANTE D’OPTIMALITÉ LOCALE 145

une direction d de \mathbb{R}^n donnée à partir d’un point x^0 de \mathbb{R}^n donné. Pour $s \in \mathbb{R}$, posons $g(s) = f(x^0 + sd)$. On obtient alors : $g'(s) = d^t \nabla f(x^0 + sd)$ et $g'(0) = d^t \nabla f(x^0)$.

8.3.3 Matrice hessienne

Si maintenant f admet des dérivées partielles d’ordre 2 en x , on pose :

$$\nabla^2 f(x) = \nabla (\nabla f(x)^t) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2}(x) & \dots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(x) \\ \frac{\partial^2 f}{\partial x_2 \partial x_1}(x) & \dots & \frac{\partial^2 f}{\partial x_2 \partial x_n}(x) \\ \dots & \dots & \dots \\ \frac{\partial^2 f}{\partial x_n \partial x_1}(x) & \dots & \frac{\partial^2 f}{\partial x_n^2}(x) \end{pmatrix} ;$$

$\nabla^2 f(x)$ s’appelle la *matrice hessienne* de f en x .

Si f est une fonction de classe C^2 (autrement dit, f admet des dérivées partielles à l’ordre 2 continues), la matrice hessienne de f est une matrice symétrique (théorème de Schwarz).

Si f est de classe C^2 en x^0 , on peut écrire la formule de Taylor d’ordre 2 : $f(x) = f(x^0) + (x - x^0)^t \nabla f(x^0) + \frac{1}{2}(x - x^0)^t \nabla^2 f(x^0)(x - x^0) + ||x - x^0||^2 \varepsilon(x)$, où $\varepsilon(x)$ tend vers 0 quand x tend vers x^0 .

8.4 Condition nécessaire et condition suffisante d’optimalité locale

On suppose ici que f est une fonction de \mathbb{R}^n dans \mathbb{R} de classe C^2 . On rappelle les définitions suivantes :

Définition 8.8. Soit M une matrice réelle carrée symétrique.

- M est *positive* si on a : $\forall h \in \mathbb{R}^n, h^t M h \geq 0$,
- M est *définie positive* si on a : $\forall h \in \mathbb{R}^n \setminus \{0\}, h^t M h > 0$.

Une matrice réelle carrée symétrique est positive si et seulement si ses valeurs propres sont positives ou nulles. Elle est définie positive si et seulement si ses valeurs propres sont strictement positives.

Théorème 8.9. (condition nécessaire d’optimalité) Si f admet un minimum local en x^* , alors :

1. $\nabla f(x^*) = 0$
2. $\nabla^2 f(x^*)$ est une matrice positive.

Preuve. D’après le développement de Taylor à l’ordre 1 en x^* , on a :

$$f(x) = f(x^*) + (x - x^*)^t \nabla f(x^*) + \|x - x^*\| \varepsilon(x),$$

où $\varepsilon(x)$ tend vers 0 quand x tend vers x^* . En particulier, en choisissant $x = x^* - s \nabla f(x^*)$, avec $s \in \mathbb{R}$, on obtient :

$$f(x) - f(x^*) = -s \|\nabla f(x^*)\|^2 + s \varepsilon_1(s) = s \left(-\|\nabla f(x^*)\|^2 + \varepsilon_1(s) \right),$$

où $\varepsilon_1(s)$ tend vers 0 quand s tend vers 0. Pour s positif, $f(x) - f(x^*)$ est du signe de $-\|\nabla f(x^*)\|^2 + \varepsilon_1(s)$. Si on a $\nabla f(x^*) \neq 0$, alors, pour s positif petit, $f(x) - f(x^*)$ est du signe de $-\|\nabla f(x^*)\|^2$ et il existe dans tout voisinage de x^* des points x vérifiant $f(x) < f(x^*)$, contradiction avec l’optimalité locale de x^* . D’où le premier énoncé.

Supposons maintenant qu’il existe $h \in \mathbb{R}^n$ tel qu’on ait la relation : $h^t \nabla^2 f(x^*) h < 0$. On a alors, d’après le développement de Taylor d’ordre 2 :

$$f(x^* + sh) - f(x^*) = s^2 \left(\frac{1}{2} h^t \nabla^2 f(x^*) h + \varepsilon_2(s) \right),$$

8.4. CONDITION NÉCESSAIRE ET CONDITION SUFFISANTE D’OPTIMALITÉ LOCALE 147

où $\varepsilon_2(s)$ tend vers 0 quand s tend vers 0. Pour s assez petit, la différence $f(x^* + sh) - f(x^*)$ serait négative, ce qui contredit l’hypothèse sur x^* . \diamond

Théorème 8.10. (condition suffisante d’optimalité) Si une fonction f vérifie en x^* :

1. $\nabla f(x^*) = 0$
2. $\nabla^2 f(x^*)$ est une matrice définie positive

alors f admet un minimum local en x^* .

Preuve. La matrice $\nabla^2 f(x^*)$ étant définie positive, il existe $a > 0$ tel que :

$$\forall h \in \mathbb{R}^n, h^t \nabla^2 f(x^*) h \geq a \|h\|^2.$$

En effet, plaçons-nous sur la sphère S de centre 0 et de rayon 1 et définissons a par $a = \inf\{h^t \nabla^2 f(x^*) h \text{ pour } h \in S\}$. La sphère étant un compact, la valeur a est atteinte : $\exists h_0 \in S$ avec $a = h_0^t \nabla^2 f(x^*) h_0 > 0$. On en déduit aisément le résultat précédent.

Soit $x \in \mathbb{R}^n$. Appliquons la formule de Taylor à l’ordre 2 en posant $h = x - x^*$:

$$f(x) - f(x^*) = f(x^* + h) - f(x^*) = \frac{1}{2} h^t \nabla^2 f(x^*) h + \|h\|^2 \varepsilon(h) \geq \|h\|^2 \left(\frac{a}{2} + \varepsilon(h) \right),$$

où $\varepsilon(h)$ tend vers 0 quand h tend vers 0, ce qui montre le théorème car, pour h de norme assez petite, $\frac{a}{2} + \varepsilon(h)$ est du signe de a , c’est-à-dire positif. On a donc $f(x) \geq f(x^*)$ quand x tend vers x^* : x^* est un minimum local de f . \diamond

8.5 Fonctions convexes

Définition 8.11. On dit qu’une *partie* de \mathbb{R}^n est *convexe* si elle contient tout segment joignant deux quelconques de ses points.

Définition 8.12. On dit qu’une fonction f définie sur une partie convexe de \mathbb{R}^n et à valeurs réelles est *convexe* si, pour tout x et tout y de son domaine de définition et pour tout λ de $]0, 1[$, on a : $f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$. Si cette inégalité est stricte, on dit que f est *strictement convexe*. On dit qu’une fonction f est *concave* si son opposée est convexe.

Dans toute la partie 8.5, on suppose que f est définie sur un ouvert convexe Ω de \mathbb{R}^n .

Théorème 8.13. Si f est une fonction convexe et admet des dérivées partielles, alors f admet un minimum global en x^* si et seulement si on a $\nabla f(x^*) = 0$.

Preuve. D’après le théorème 8.9., si x^* est un minimum, alors on a $\nabla f(x^*) = 0$. Montrons la réciproque : si $\nabla f(x^*)$ vaut 0, alors f admet un minimum global en x^* . Soit $x \in \Omega$. Pour $s \in [0, 1]$, posons $g(s) = f(x^* + s(x - x^*))$. On constate que l’on a $g(0) = f(x^*)$, $g(1) = f(x)$ et $g'(0) = (x - x^*)^t \nabla f(x^*) = 0$. De plus, g est une fonction convexe (car f l’est). La dérivée d’une fonction convexe étant croissante, on a, pour $s \in [0, 1]$, $g'(s) \geq g'(0) = 0$; donc g est croissante pour $s \geq 0$, d’où $g(1) \geq g(0)$: f admet bien un minimum global en x^* . \diamond

Théorème 8.14. Si f est convexe et admet un minimum local en x^* , alors f admet un minimum global en x^* .

Preuve. Si f admet un minimum local en x^* , alors $\nabla f(x^*) = 0$. Comme en outre f est convexe, le théorème précédent permet de

conclure que f admet un minimum global en x^* .

◇

Le théorème suivant sera généralisé, dans le cadre de l’optimisation avec contraintes, par le théorème 9.16., page 180, qui en donnera aussi la preuve.

Théorème 8.15. Si f est strictement convexe, f admet au plus un minimum global.

On admettra le théorème suivant.

Théorème 8.16. Si f est deux fois continûment dérivable, les propositions suivantes sont équivalentes.

1. La fonction f est convexe (respectivement strictement convexe).
2. Pour tout x^0 de Ω , l’hyperplan tangent en $(x^0, f(x^0))$ à la surface d’équation $x_{n+1} = f(x)$ est en dessous (respectivement strictement en dessous, sauf en x^0) de cette surface : pour tout x de Ω , on a $f(x) \geq f(x^0) + (\nabla f(x^0))^t(x - x^0)$ (respectivement, pour tout $x \neq x^0$, $f(x) > f(x^0) + (\nabla f(x^0))^t(x - x^0)$).
3. Pour tout x de Ω , $\nabla^2 f(x)$ est positive (respectivement définie positive).

8.6 Fonctions quadratiques

Soient A une matrice réelle symétrique d’ordre n , b un vecteur-colonne d’ordre n et c un nombre réel. L’application q de \mathbb{R}^n dans \mathbb{R} définie par :

$$q(x) = c + b^t x + \frac{1}{2} x^t A x$$

s’appelle *fonction quadratique* ou aussi *forme quadratique*.

Remarque. La partie polynomiale du développement de Taylor d'ordre 2 d'une fonction f est la fonction quadratique q telle que la surface d'équation $x_{n+1} = q(x)$ soit « la plus proche » de la surface d'équation $x_{n+1} = f(x)$ au voisinage du point considéré.

On a, en utilisant les formules données dans la partie 8.3.2, page 144 :

$$\nabla q(x) = \nabla(x^t)b + \frac{1}{2}[\nabla(x^t)Ax + \nabla((Ax)^t)x].$$

Or, $\nabla(x^t)$ est la matrice identité. On a de plus les égalités suivantes :

$$\nabla((Ax)^t) = \nabla(x^t A^t) = \nabla(x^t)A^t = A^t = A.$$

D'où l'expression du gradient : $\nabla q(x) = b + Ax$.

Par ailleurs : $\nabla^2 q(x) = \nabla((\nabla q(x))^t) = \nabla(b^t + x^t A^t) = A^t = A$. On en déduit que q est convexe (respectivement strictement convexe) si et seulement si A est positive (respectivement définie positive) ; de plus, si q est strictement convexe, A est inversible et q admet un minimum global unique atteint en $x = -A^{-1}b$, point où $\nabla q(x)$ s'annule.

Les dérivées d'ordre au moins 3 de q sont nulles. Une fonction quadratique coïncide avec son développement de Taylor à l'ordre 2.

8.7 Méthodes de descente

On suppose jusqu'à la fin du chapitre que l'on a $\Omega = \mathbb{R}^n$.

8.7.1 Généralités

Même si on s'intéresse le plus souvent à des extrema globaux, on cherchera en général des extrema locaux, quitte à examiner ensuite (si possible) s'il s'agit d'extrema globaux.

Quand nous considérerons des fractions dans ce qui suit, nous supposerons que les dénominateurs sont non nuls (les adaptations étant immédiates dans le cas contraire). Pour déterminer un point où une fonction f atteint un minimum local, les méthodes consistent très souvent à construire une suite $x^0, x^1, \dots, x^k, \dots$ qui doit converger vers un point x^* vérifiant une condition nécessaire d'optimalité. Cette condition (souvent $\nabla f(x^*) = 0$) n'est en général pas suffisante et le comportement de f au voisinage de x^* doit donc faire l'objet d'une étude supplémentaire (pouvant porter entre autres sur la matrice hessienne de f en x^*).

On appelle *méthode de descente* toute méthode où, à chaque étape, on pose $x^{k+1} = x^k + s_k d^k$, avec $s_k \in \mathbb{R}_+$ et où d^k est une direction de \mathbb{R}^n qui vérifie $(d^k)^t \nabla f(x^k) < 0$. Cette dernière condition signifie que la fonction $s \mapsto f(x^k + s d^k)$ a une dérivée négative pour $s = 0$: partant de x^k dans la direction d^k , f décroît (« on descend ») ; une telle direction est dite *direction de descente*. La différence entre les diverses méthodes de descente porte sur le choix de s_k et de d^k , choix qui doit au moins assurer $f(x^{k+1}) \leq f(x^k)$.

8.7.2 Vitesse de convergence

Lorsque la convergence d'une méthode de descente a été établie, une qualité importante de cette méthode est sa *vitesse de convergence*.

- Si on a $\frac{\|x^{k+1} - x^*\|}{\|x^k - x^*\|} \leq \alpha < 1$ pour k assez grand, on dit que la convergence est *linéaire de taux α* .
- Si $\frac{\|x^{k+1} - x^*\|}{\|x^k - x^*\|}$ tend vers 0 quand k tend vers l’infini, on dit que la convergence est *superlinéaire*.
- Si $\frac{\|x^{k+1} - x^*\|}{\|x^k - x^*\|^\gamma}$ est borné, avec $\gamma > 1$, on dit que la convergence est *superlinéaire d’ordre γ* . Dans le cas $\gamma = 2$, on dit que la convergence est *quadratique*.

8.7.3 Méthodes de gradient

Il s’agit de méthodes de descente qui s’appliquent à des fonctions dérivables et qui utilisent l’idée qui suit.

Soient d un vecteur de \mathbb{R}^n et x^k un point de \mathbb{R}^n avec $\nabla f(x^k) \neq 0$. Posons, pour $s \in \mathbb{R}$: $g(s) = f(x^k + sd)$.

On dit que d est une *direction de descente* si on a $g'(0) < 0$. Nous avons vu la relation $g'(0) = d^t \nabla f(x^k)$. D’où, en notant θ l’angle entre $\nabla f(x^k)$ et d :

$$g'(0) = \|\nabla f(x^k)\| \times \|d\| \times \cos \theta.$$

En choisissant d unitaire (ou plus généralement de norme majorée par une constante), $g'(0)$ est minimum pour $\cos \theta = -1$, c’est-à-dire si d est donnée par l’opposé du gradient :

$$d = - \frac{\nabla f(x^k)}{\|\nabla f(x^k)\|}.$$

Cette dernière direction donne ce qu’on appelle la *direction de plus grande pente descendante*. C’est ce choix qui est fait dans les méthodes de gradient.

8.7.4 Méthode de la plus forte pente à pas fixe

Dans cette méthode de gradient, l’amplitude du déplacement, appelée *pas*, dans la direction $-\nabla f(x^k)$ est constante. La valeur du pas est donc fixée à l’avance. L’algorithme peut s’écrire de la façon suivante :

- définir une constante λ strictement positive pour la longueur du pas
- choisir un point de départ x^0
- $k \leftarrow 0$
- répéter

$$\star x^{k+1} \leftarrow x^k - \lambda \frac{\nabla f(x^k)}{\|\nabla f(x^k)\|}$$

$$\star k \leftarrow k + 1$$

tant qu’un test d’arrêt donné n’est pas vérifié.

8.7.5 Méthode de la plus forte pente à pas optimal

Il s’agit d’une méthode de gradient dans laquelle on choisit $d^k = -\nabla f(x^k)$ pour avoir la plus forte pente⁴. On pose ensuite $g(s) = f(x^k - s\nabla f(x^k))$ et on cherche s_k de façon à minimiser g pour $s \geq 0$ (si un tel s_k existe). On est alors ramené à un problème d’optimisation unidimensionnelle.

La méthode de la plus forte pente à pas optimal peut s’écrire de la façon suivante (l’exercice à la fin du chapitre, page 159, donne une illustration de cette méthode) :

- choisir un point de départ x^0

4. Cette méthode de descente de plus forte pente est généralement attribuée à L. A. Cauchy, Méthode générale pour la résolution des systèmes d’équations simultanées, *Comptes rendus hebdomadaires des séances de l’Académie des sciences* 25, 1847, 536-538.

- $k \leftarrow 0$
 - répéter
 - ★ $d^k \leftarrow -\nabla f(x^k)$
 - ★ définir la fonction g sur $[0, +\infty[$ par $g(s) = f(x^k + sd^k)$
 - ★ si g tend asymptotiquement vers $-\infty$, conclure que f n'a pas de minimum fini et s'arrêter
 - sinon
 - ▷ si g est décroissante et tend asymptotiquement vers une limite finie : $x^{k+1} \leftarrow x^k + \lambda d^k$ où λ est une constante positive ($\lambda > 0$)
 - sinon (g admet un minimum local)
 - ◇ si on peut déterminer une valeur de s_k correspondant à un minimum global de g : $x^{k+1} \leftarrow x^k + s_k d^k$
 - sinon, on cherche la plus petite valeur positive de s_k correspondant à un minimum local de g ; en pratique, on n'est pas sûr d'obtenir une telle plus petite valeur de s_k , mais il faut cependant avoir $g(s_k) < g(0)$
 - ★ $k \leftarrow k + 1$
- tant qu'un test d'arrêt donné n'est pas vérifié.

Le test d'arrêt peut être l'un des suivants :

- on a épuisé un nombre d'itérations fixé à l'avance ;
- le gradient en x^k est suffisamment proche de 0 : $\|\nabla f(x^k)\| \leq \varepsilon$, où ε est un paramètre donné ;
- la suite x^k est « presque » stationnaire : $f(x^k) - f(x^{k+1}) \leq \varepsilon$ ou $\|x^{k+1} - x^k\| \leq \varepsilon$, où ε est un paramètre donné.

On peut aussi exiger que l’un de ces tests soit vérifié sur plusieurs itérations ou que plusieurs tests soient satisfaits simultanément.

On peut montrer que, si la fonction f est de classe C^1 et tend vers l’infini quand $\|x\|$ tend vers l’infini, cet algorithme converge vers un point stationnaire (point où le gradient s’annule).

L’inconvénient de cette méthode est que la vitesse de convergence peut être très faible (linéaire avec un taux proche de 1). Cette lenteur peut s’expliquer de la façon suivante : l’égalité $\frac{d}{ds}[f(x^k - s\nabla f(x^k))](s_k) = 0$ s’écrit : $[\nabla f(x^k)]^t \nabla f(x^{k+1}) = 0$; les directions de déplacement successives sont orthogonales. Sur la figure 8.2, on a représenté quelques courbes de niveau et les déplacements. Il y a *convergence en zig-zag*.

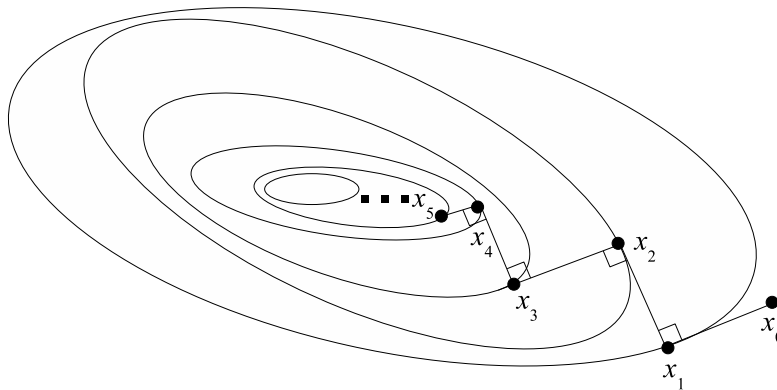


FIGURE 8.2 – Convergence en zig-zag.

8.7.6 Méthode de la plus forte pente accélérée

La méthode de la plus forte pente accélérée est une méthode de descente qui s’appuie sur la méthode de la plus forte pente à pas optimal et qui accélère celle-ci en essayant d’éviter la convergence en zig-zag.

Soit p un entier fixé. À partir de x^k , on effectue p itérations de la méthode de la plus forte pente à pas optimal ; on obtient un point y^k et on pose $d^k = y^k - x^k$. Le point x^{k+1} est le point où la fonction $f(x^k + sd^k)$ admet un minimum pour $s > 0$.

La figure 8.3 illustre cette méthode dans le cas $p = 2$. Pour $p = 1$, on retrouve la méthode de la plus forte pente à pas optimal.

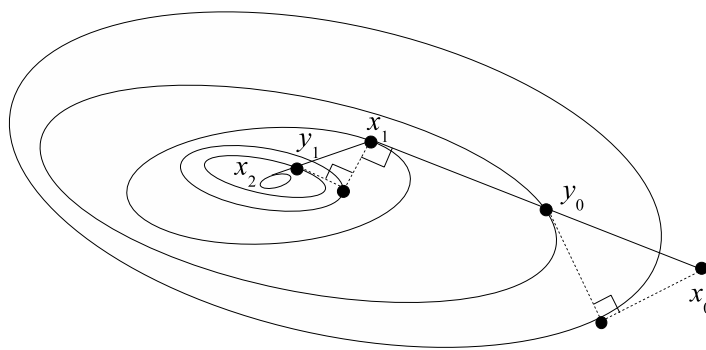


FIGURE 8.3 – Convergence accélérée.

8.8 Méthode de Newton

On suppose ici que f est de classe C^3 .

Au voisinage d'un point x^k , on approche f par la fonction quadratique q donnée par la formule de Taylor d'ordre 2 :

$$q(x) = f(x^k) + (x - x^k)^t \nabla f(x^k) + \frac{1}{2} (x - x^k)^t \nabla^2 f(x^k) (x - x^k).$$

On peut alors choisir pour x^{k+1} le point, s'il existe, qui minimise q ; pour que ce point minimisant q existe, il est suffisant que $\nabla^2 f(x^k)$ soit définie positive ; x^{k+1} est alors déterminé par l'équation $\nabla q(x^{k+1}) = 0$, qui s'écrit :

$$\nabla f(x^k) + \nabla^2 f(x^k)(x^{k+1} - x^k) = 0,$$

d'où :

$$x^{k+1} = x^k - [\nabla^2 f(x^k)]^{-1} \nabla f(x^k).$$

L'exercice proposé plus loin illustre la méthode de Newton.

Proposition 8.17. Si x^0 est choisi suffisamment proche d'un minimum local x^* en lequel la matrice hessienne de f est définie positive, alors la suite (x^k) a une convergence quadratique vers x^* .

Preuve. On considère une norme vectorielle $\|\cdot\|$ et la norme matricielle subordonnée $\|\cdot\|$ à celle-ci (voir l'annexe ??, page ??). On va établir une condition suffisante portant sur x^0 pour assurer une convergence quadratique de la suite (x^k) construite à partir de x^0 par la méthode de Newton. Pour cela, on considère les éléments suivants.

- On sait que $\nabla^2 f(x^*)$ est définie positive. Par continuité de la fonction $\nabla^2 f$, il existe une boule B_1 de centre x^* et de rayon r_1 sur laquelle $\nabla^2 f(x)$ est définie positive et donc inversible. On note alors M un majorant strictement positif de $\|(\nabla^2 f)^{-1}\|$ sur B_1 (un tel majorant existe puisque $(\nabla^2 f)^{-1}$ est continue et que B_1 est un compact).
- En utilisant le fait que f est de classe C^3 , la formule de Taylor avec reste intégral montre qu'il existe une constante N strictement positive et une fonction $\phi(a, b)$ pour laquelle on a, si a et b sont deux points de B_1 :
 - ★ $\nabla f(b) = \nabla f(a) + \nabla^2 f(a)(b - a) + \phi(a, b)\|b - a\|^2$
 - ★ $\|\phi(a, b)\| \leq N.$ (1)
- On pose $M' = MN > 0$.
- On considère un réel r vérifiant simultanément $r \leq r_1$ et $r < \frac{1}{M'}$; on appelle B la boule centrée en x^* et de rayon r (on notera que B est incluse dans B_1).

On fait l'hypothèse que x^0 est dans B . On va montrer par récurrence sur k que la suite (x^k) est toute entière dans B . C'est vrai pour $k = 0$ et on suppose que c'est vrai pour $k \geq 0$.

On a : $\nabla f(x^*) - \nabla f(x^k) = \nabla^2 f(x^k)(x^* - x^k) + \phi(x^k, x^*)||x^* - x^k||^2$.

En utilisant $\nabla f(x^*) = 0$ (conséquence de la minimalité de x^*) :

$$\nabla^2 f(x^k)(x^k - x^*) = \nabla f(x^k) + \phi(x^k, x^*)||x^k - x^*||^2.$$

La matrice $\nabla^2 f(x^k)$ étant inversible (puisque l'on a $||x^k - x^*|| \leq r_1$), on obtient, en multipliant à gauche les deux membres par $[\nabla^2 f(x^k)]^{-1}$:

$$x^k - x^* = [\nabla^2 f(x^k)]^{-1} \nabla f(x^k) + [\nabla^2 f(x^k)]^{-1} \phi(x^k, x^*)||x^k - x^*||^2. \quad (2)$$

Par ailleurs :

$$x^{k+1} - x^* = (x^{k+1} - x^k) + (x^k - x^*). \quad (3)$$

Par construction :

$$x^{k+1} - x^k = -[\nabla^2 f(x^k)]^{-1} \nabla f(x^k). \quad (4)$$

En utilisant les égalités (2), (3) et (4), on obtient :

$$\begin{aligned} x^{k+1} - x^* &= -[\nabla^2 f(x^k)]^{-1} \nabla f(x^k) + [\nabla^2 f(x^k)]^{-1} \nabla f(x^k) \\ &\quad + [\nabla^2 f(x^k)]^{-1} \phi(x^k, x^*)||x^k - x^*||^2 \\ &= [\nabla^2 f(x^k)]^{-1} \phi(x^k, x^*)||x^k - x^*||^2. \end{aligned}$$

D'où : $||x^{k+1} - x^*|| \leq ||[\nabla^2 f(x^k)]^{-1}|| ||\phi(x^k, x^*)|| ||x^k - x^*||^2$,
ce qui entraîne, en exploitant la propriété (1) et l'inégalité $r \leq r_1$:

$$||x^{k+1} - x^*|| \leq MN||x^k - x^*||^2 = M'||x^k - x^*||^2 = (M'||x^k - x^*||)||x^k - x^*||.$$

Comme x^k est dans la boule B , $||x^k - x^*|| \leq r < \frac{1}{M'}$; d'où $M'||x^k - x^*|| < 1$; on a donc : $||x^{k+1} - x^*|| < ||x^k - x^*|| \leq r$; x^{k+1} est aussi dans la boule B . On a ainsi établi que toute la suite (x^k) est dans B .

8.9. EXERCICE

159

Posons $\alpha = M' \|x^0 - x^*\|$. On a $\|x^0 - x^*\| \leq r < \frac{1}{M'}$, d'où $\alpha < 1$.

On a aussi : $M' \|x^{k+1} - x^*\| \leq (M' \|x^k - x^*\|)^2$; on obtient par récurrence :

$$M' \|x^k - x^*\| \leq (M' \|x^0 - x^*\|)^{2^k} = \alpha^{2^k},$$

ou encore : $\|x^k - x^*\| \leq \frac{\alpha^{2^k}}{M'}$.

La suite x^k converge donc vers x^* .

Enfin, la relation $\|x^{k+1} - x^*\| \leq M' \|x^k - x^*\|^2$ montre que la convergence est quadratique. \diamond

8.9 Exercice

Énoncé. On s'intéresse au minimum de la fonction f définie sur \mathbb{R}^2 par :

$$f(x, y) = e^{x+y} + x^2 + 2y^2.$$

Q1. Appliquer trois itérations de la méthode de la plus forte pente à pas optimal à partir du point $(0, 0)$.

Q2. Appliquer deux itérations de la méthode de Newton à partir du point $(0, 0)$.

Corrigé. La fonction f est de classe C^∞ . Commençons par déterminer le gradient et la matrice hessienne de f :

$$\nabla f(x, y) = \begin{pmatrix} e^{x+y} + 2x \\ e^{x+y} + 4y \end{pmatrix}, \quad \nabla^2 f(x, y) = \begin{pmatrix} e^{x+y} + 2 & e^{x+y} \\ e^{x+y} & e^{x+y} + 4 \end{pmatrix}.$$

Le déterminant de la matrice hessienne (produit des valeurs propres) ainsi que sa trace (somme des valeurs propres) étant strictement positifs, les valeurs propres de $\nabla^2 f(x, y)$ sont strictement

positives et $\nabla^2 f(x, y)$ est définie positive : f est donc strictement convexe.

On en déduit que tout minimum local est global, et une condition nécessaire et suffisante pour que (x^*, y^*) soit un minimum est $\nabla f(x^*, y^*) = 0$. Comme par ailleurs f tend vers l’infini à l’infini, f admet un minimum global.

Q1. Appliquons la méthode du gradient à pas optimal. On part du point $P^0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$; d’où $\nabla f(P^0) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ et $d^0 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$. On cherche s de sorte que $g(s) = f(P^0 + sd^0) = f(-s, -s) = e^{-2s} + 3s^2$ soit minimum. On minimise g par exemple par dichotomie et on trouve $s \simeq 0,216$. D’où :

$$P^1 \simeq P^0 + 0,216d^0 = \begin{pmatrix} -0,216 \\ -0,216 \end{pmatrix}; \nabla f(P^1) \simeq \begin{pmatrix} 0,216 \\ -0,216 \end{pmatrix}; d^1 \simeq \begin{pmatrix} -0,216 \\ 0,216 \end{pmatrix}.$$

On constate que d^1 est orthogonal à d^0 . On pose :

$$\begin{aligned} g(s) &= f(P^1 + sd^1) \\ &\simeq f(-0,216(1+s); -0,216(1-s)) \\ &\simeq e^{-2 \times 0,216} + (0,216)^2 [(1+s)^2 + 2(1-s)^2] \\ &\simeq e^{-2 \times 0,216} + (0,216)^2 h(s) \end{aligned}$$

avec $h(s) = [(1+s)^2 + 2(1-s)^2] = 3s^2 - 2s + 3$.

Le minimum de h est atteint pour $s = 1/3$. D’où :

$$\begin{aligned} P^2 &\simeq \begin{pmatrix} -0,216(1+1/3) \\ -0,216(1-1/3) \end{pmatrix} = \begin{pmatrix} -0,288 \\ -0,144 \end{pmatrix}; \nabla f(P^2) \simeq \\ &\begin{pmatrix} 0,0732 \\ 0,0732 \end{pmatrix}; \end{aligned}$$

8.9. EXERCICE

161

$d^2 \simeq \begin{pmatrix} -0,0732 \\ -0,0732 \end{pmatrix}$ (on constate de nouveau que d^2 est orthogonal à d^1).

On considère maintenant :

$$\begin{aligned} g(s) &= f(P^2 + sd^2) \\ &\simeq f(-0,288 - 0,0732s ; -0,144 - 0,0732s) \\ &\simeq e^{-0,432-0,1464s} + (0,288 + 0,0732s)^2 + 2(0,144 + 0,0732s)^2. \end{aligned}$$

On minimise g par dichotomie et on trouve $s \simeq 0,2339$; d'où :
 $P^3 \simeq \begin{pmatrix} -0,305 \\ -0,161 \end{pmatrix}$.

On peut continuer ainsi pour avoir plus de précision ; on obtiendrait à la fin le point $P^* \simeq \begin{pmatrix} -0,3128 \\ -0,1564 \end{pmatrix}$ (point où le gradient de f s'annule), avec un minimum de f environ égal à 0,7723.

Si on considère les distances euclidiennes d_E entre les points déterminés par la méthode et P^* , on obtient : $d_E(P^0, P^*) \simeq 0,3497$, $d_E(P^1, P^*) \simeq 0,1137$, $d_E(P^2, P^*) \simeq 0,0277$, $d_E(P^3, P^*) \simeq 0,0091$.

Q2. Appliquons la méthode de Newton. On part du point $P^0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$.

En posant $P^1 = \begin{pmatrix} x^1 \\ y^1 \end{pmatrix}$, la première itération consiste à résoudre le système $\nabla^2 f(P^0)(P^1 - P^0) = -\nabla f(P^0)$, c'est-à-dire

$$\begin{cases} 3x^1 + y^1 = -1 \\ x^1 + 5y^1 = -1 \end{cases}, \text{ dont la solution est } P^1 = \begin{pmatrix} -2/7 \\ -1/7 \end{pmatrix} \simeq \begin{pmatrix} -0,2857 \\ -0,1429 \end{pmatrix};$$
on a ici $d_E(P^1, P^*) \simeq 0,0303$.

L'itération suivante consiste à résoudre $\nabla^2 f(P^1)(P^2 - P^1) = -\nabla f(P^1)$,
 c'est-à-dire $\begin{cases} 2,6514(x^2 + 0,2857) + 0,6514(y^2 + 0,1429) \simeq -0,0800 \\ 0,6514(x^2 + 0,2857) + 4,6514(y^2 + 0,1429) \simeq -0,0800 \end{cases}$
 en posant $P^2 = \begin{pmatrix} x^2 \\ y^2 \end{pmatrix}$. On obtient $P^2 \simeq \begin{pmatrix} -0,3126 \\ -0,1563 \end{pmatrix}$, avec
 $d_E(P^2, P^*) \simeq 0,0002$.

Cet exemple illustre le fait que la méthode de Newton converge généralement plus vite que la méthode du gradient à pas optimal.

Chapitre 9

Optimisation non linéaire avec contraintes

9.1 Généralités

Soit Ω un ouvert de \mathbb{R}^n (dans cet ouvrage, on aura souvent $\Omega = \mathbb{R}^n$). On considère des fonctions g_i ($1 \leq i \leq m$) et h_j ($1 \leq j \leq p$) définies et continues sur Ω et à valeurs réelles. On pose $I = \{1, \dots, m\}$ et $J = \{1, \dots, p\}$. Soit X l'ensemble des éléments de Ω vérifiant :

$$\begin{cases} \text{pour } i \in I, & g_i(x) \leq 0 \\ \text{pour } j \in J, & h_j(x) = 0. \end{cases}$$

Si on considère une fonction continue de \mathbb{R}^n dans \mathbb{R} , l'image réciproque par cette fonction d'un fermé de \mathbb{R} (ici, l'intervalle $]-\infty, 0]$ ou l'intervalle $[0, 0]$) est un fermé de \mathbb{R}^n . Par ailleurs, l'intersection de fermés de \mathbb{R}^n est un fermé de \mathbb{R}^n . En conséquence, l'ensemble X est un fermé de \mathbb{R}^n .

On considère maintenant une fonction f définie sur l'ouvert Ω et à valeurs réelles. On s'intéresse au problème (P) :

$$\text{minimiser } f(x) \text{ pour } x \in X.$$

Les adaptations à un problème de maximisation avec contraintes sont immédiates.

Les conditions $g_i(x) \leq 0$ et $h_j(x) = 0$ s'appellent les *contraintes* du problème (P) . Tout élément x de X s'appelle *solution réalisable* et X est le *domaine réalisable* (on dit aussi *admissible*). Si, pour $i \in I$ et pour $x \in X$, on a $g_i(x) = 0$, on dit que la contrainte g_i est *saturée* ou *serrée* en x .

On supposera dans tout ce chapitre que les fonctions g_i ($i \in I$), h_j ($j \in J$) et f sont de classe C^1 sur Ω et que le domaine réalisable X est non vide. Quand on considérera une norme $\| \cdot \|$, il s'agira, sauf mention contraire, de la norme euclidienne dans \mathbb{R}^n .

Théorème 9.1. Si le domaine réalisable X est borné, le problème (P) admet un minimum global et celui-ci est atteint par au moins un élément de X .

Preuve. Avec les hypothèses, le domaine réalisable est un fermé borné non vide de \mathbb{R}^n , c'est-à-dire un compact non vide de \mathbb{R}^n . Le résultat découle immédiatement du théorème 8.7., page 143. \diamond

Définition 9.2. La fonction f est dite *coercive* si, pour tout réel M , il existe un réel r tel que, pour $x \in \Omega$ vérifiant $\|x\| \geq r$, on ait $f(x) \geq M$ (autrement dit, la fonction f tend vers l'infini si x tend vers l'infini en restant dans Ω).

Théorème 9.3. Si la fonction f est coercive, alors le problème (P) admet un minimum global et celui-ci est atteint par au moins un élément de X .

Preuve. Soit x une solution réalisable. Puisque f est coercive, il existe une boule fermée B de \mathbb{R}^n de rayon non nul centrée sur l'origine telle que, pour tout x' dans Ω et non dans B , on ait

$f(x') > f(x)$. L'ensemble $B \cap X$ étant un ensemble fermé, borné et non vide de \mathbb{R}^n , la fonction f admet un minimum global sur $B \cap X$ en un point x^* . Ce point est aussi une solution optimale pour le problème (P) . \diamond

Définition 9.4. On dit qu'une *direction* d est *admissible* en $x^0 \in X$ s'il existe une fonction ϕ de \mathbb{R} dans \mathbb{R}^n vérifiant :

1. $\phi(0) = x^0$
2. pour tout $t > 0$ assez petit, $\phi(t) \in X$
3. la dérivée à droite de ϕ en 0 est d .

Autrement dit, une direction d est admissible en x^0 s'il y a moyen de ne pas sortir du domaine réalisable quand on part de x^0 en suivant d tangentiellement.

Soit $x^0 \in X$. On note $A(x^0)$ l'ensemble des directions admissibles en x^0 ; on pose $I_0(x^0) = \{i \in I \text{ vérifiant } g_i(x^0) = 0\}$.

Proposition 9.5. Si d est une direction admissible en $x^0 \in X$, alors :

1. pour $i \in I_0(x^0)$, $d^t \nabla g_i(x^0) \leq 0$
2. pour $j \in J$, $d^t \nabla h_j(x^0) = 0$.

Preuve. Soit ϕ une fonction correspondant à la définition 9.4.. Appliquons la formule de Taylor à l'ordre 1.

1. Si on a $g_i(x^0) = 0$, alors : $g_i(\phi(s)) = s d^t \nabla g_i(x^0) + s \varepsilon(s)$ où $\varepsilon(s) \rightarrow 0$ quand $s \rightarrow 0$. Pour $s > 0$ assez petit, on a $g_i(\phi(s)) \leq 0$ et donc : $d^t \nabla g_i(x^0) + \varepsilon(s) \leq 0$, ce qui donne le résultat en passant à la limite quand $s \rightarrow 0$.
2. De même : $h_j(\phi(s)) = h_j(x^0) + s d^t \nabla h_j(x^0) + s \varepsilon(s)$ où $\varepsilon(s) \rightarrow 0$ quand $s \rightarrow 0$. Pour $s > 0$ assez petit, $h_j(\phi(s)) = 0$ et $h_j(x^0) = 0$; on a donc pour $s > 0$ assez petit : $d^t \nabla h_j(x^0) +$

$\varepsilon(s) = 0$, ce qui donne le résultat en passant à la limite quand $s \rightarrow 0$. \diamond

On note $B(x^0)$ l'ensemble des directions d vérifiant :

- pour $i \in I_0(x^0)$, $d^t \nabla g_i(x^0) \leq 0$
- $j \in J$, $d^t \nabla h_j(x^0) = 0$.

La proposition 9.5. se réécrit : $A(x^0) \subseteq B(x^0)$. L'exemple de la figure 9.1 donné plus loin (page 170) illustre le cas d'une direction d appartenant à $B(x^0) \setminus A(x^0)$.

Définition 9.6. On dit que les *contraintes* sont *qualifiées* en $x^0 \in X$ si toute direction dans $B(x^0)$ est limite d'une suite de directions de $A(x^0)$.

Les propositions suivantes donnent des conditions suffisantes pour que des contraintes soient qualifiées.

Proposition 9.7. Si :

- les fonctions g_i sont convexes,
- les fonctions h_j sont affines,
- il existe $\tilde{x} \in X$ avec, pour tout $i \in I$, $g_i(\tilde{x}) < 0$,

alors les contraintes sont qualifiées en tout point de X .

Proposition 9.8. On suppose que, pour $j \in J$, les fonctions h_j sont affines. Si, en $x^0 \in X$, les gradients

- $\nabla g_i(x^0)$ pour $i \in I_0(x^0)$
- $\nabla h_j(x^0)$ pour $j \in J$

sont linéairement indépendants, alors les contraintes sont qualifiées en x^0 .

Avant de prouver ces propositions, on établit deux lemmes :

Lemme 9.9. On suppose que, pour $j \in J$, les fonctions h_j sont affines. Soient $x^0 \in X$ et d une direction vérifiant :

- pour $i \in I_0(x^0)$, $d^t \nabla g_i(x^0) < 0$
- pour $j \in J$, $d^t \nabla h_j(x^0) = 0$.

Alors d est une direction admissible en x^0 .

Preuve. Pour $s \geq 0$, on pose : $\phi(s) = x^0 + sd$. On a $\phi(0) = x^0$ et $\phi'(0) = d$: les points 1 et 3 de la définition d’une direction admissible sont satisfaits.

Pour j appartenant à J , puisque les fonctions h_j sont affines, on peut écrire : $h_j(\phi(s)) = h_j(x^0) + sd^t \nabla h_j(x^0)$. Par hypothèse sur x^0 , on a $h_j(x^0) = 0$ et, par hypothèse sur d , on a $d^t \nabla h_j(x^0) = 0$. D’où $h_j(\phi(s)) = 0$.

De plus, pour $i \in I_0(x^0)$, on peut écrire :

$g_i(\phi(s)) = g_i(x^0) + s(d^t \nabla g_i(x^0) + \varepsilon(s))$, où $\varepsilon(s) \rightarrow 0$ quand $s \rightarrow 0$.

De $g_i(x^0) = 0$ et $d^t \nabla g_i(x^0) < 0$, on déduit $g_i(\phi(s)) \leq 0$ pour s positif assez petit.

Ainsi, la direction d est admissible. \diamond

Lemme 9.10. On suppose que, pour $j \in J$, les fonctions h_j sont affines. Soit $x^0 \in X$. S’il existe une direction \tilde{d} telle que :

- pour $i \in I_0(x^0)$, $\tilde{d}^t \nabla g_i(x^0) < 0$
- pour $j \in J$, $\tilde{d}^t \nabla h_j(x^0) = 0$,

alors les contraintes sont qualifiées en x^0 .

Preuve. Soit $d \in B(x^0)$ et soit \tilde{d} vérifiant les hypothèses du lemme.

Pour $\lambda \in [0, 1[$, soit $d_\lambda = \lambda d + (1 - \lambda)\tilde{d}$.

Pour $i \in I_0(x^0) : d_\lambda^t \nabla g_i(x^0) = \lambda d^t \nabla g_i(x^0) + (1-\lambda) \tilde{d}^t \nabla g_i(x^0) < 0$.

Pour $j \in J : d_\lambda^t \nabla h_j(x^0) = \lambda d^t \nabla h_j(x^0) + (1-\lambda) \tilde{d}^t \nabla h_j(x^0) = 0$.

Le lemme 9.9. indique que, pour tout $\lambda \in [0, 1[$, d_λ est une direction admissible. En considérant une suite de nombres λ_n tendant vers 1 par valeurs inférieures, on obtient une suite de directions admissibles d_{λ_n} qui tend vers d : cela montre que les contraintes sont qualifiées en x^0 . \diamond

Preuve de la proposition 9.7.. Soit $\tilde{x} \in X$ vérifiant, pour tout $i \in I$, $g_i(\tilde{x}) < 0$ et soit x^0 un point quelconque de X .

En utilisant la convexité des g_i , on a pour $i \in I_0(x^0) :$

$$0 > g_i(\tilde{x}) \geq g_i(x^0) + (\tilde{x} - x^0)^t \nabla g_i(x^0).$$

D'où $(\tilde{x} - x^0)^t \nabla g_i(x^0) < 0$, puisque l'on a $g_i(x^0) = 0$. On pose $\tilde{d} = \tilde{x} - x^0$; on a donc $\tilde{d}^t \nabla g_i(x^0) < 0$.

Pour $j \in J$, les h_j étant affines : $0 = h_j(\tilde{x}) = h_j(x^0) + \tilde{d}^t \nabla h_j(x^0)$; d'où on déduit : $\tilde{d}^t \nabla h_j(x^0) = 0$.

On utilise le lemme 9.10. : les contraintes sont qualifiées en x^0 . Comme x^0 est un point quelconque de X , on conclut que les contraintes sont qualifiées en tout point de X . \diamond

Preuve de la proposition 9.8.. On considère les deux problèmes (Q) et (R) d'optimisation linéaire ci-dessous :

$$(Q) \quad \begin{aligned} & \text{Maximiser } z = \sum_{i \in I_0(x^0)} \lambda_i \\ & \text{avec } \begin{cases} \sum_{j \in J} \mu_j \nabla h_j(x^0) - \sum_{i \in I_0(x^0)} \lambda_i \nabla g_i(x^0) = 0 \\ \text{pour } i \in I_0(x^0), \lambda_i \geq 0 \\ \text{pour } j \in J, \mu_j \in \mathbb{R} \end{cases} \end{aligned}$$

$$(R) \quad \begin{array}{l} \text{Minimiser } w = 0 \\ \text{avec } \left\{ \begin{array}{l} \text{pour } i \in I_0(x^0), d^t \nabla g_i(x^0) \leq -1 \\ \text{pour } j \in J, d^t \nabla h_j(x^0) = 0 \\ d \in \mathbb{R}^n. \end{array} \right. \end{array}$$

On peut facilement vérifier que les problèmes (Q) et (R) sont duaux l'un de l'autre (au sens de l'optimisation linéaire, voir le chapitre 6).

Le problème (Q) est réalisable puisque la solution nulle l'est ; montrons qu'il est majoré par 0. Supposons qu'il puisse prendre une valeur strictement positive. Alors, dans cette solution, au moins un λ_i ($i \in I_0(x^0)$) est non nul et les vecteurs $\nabla g_i(x^0)$ ($i \in I_0(x^0)$) et $\nabla h_j(x^0)$ ($j \in J$) sont linéairement dépendants, ce qui est contraire à l'hypothèse. Le maximum du problème (Q) existe (il vaut 0).

On utilise le théorème de la dualité (théorème 6.2, page 74) pour l'optimisation linéaire : le problème (Q) étant réalisable et borné, le problème (R) est réalisable. On note \tilde{d} une solution réalisable de (R) . Il suffit alors d'utiliser le lemme 9.10. pour conclure. \diamond

Exemple de point où les contraintes ne sont pas qualifiées.

On considère dans \mathbb{R}^2 le domaine représenté sur la figure 9.1 et défini par :

$$\left\{ \begin{array}{l} y \leq x^3 \\ x \leq 1 \\ y \geq 0. \end{array} \right.$$

On pose $g_1(x, y) = y - x^3$, $g_2(x, y) = x - 1$, $g_3(x, y) = -y$; les contraintes s'écrivent : $g_1(x, y) \leq 0$, $g_2(x, y) \leq 0$, $g_3(x, y) \leq 0$.

Au point $(0, 0)$, les contraintes g_1 et g_3 sont saturées. Or, on a :

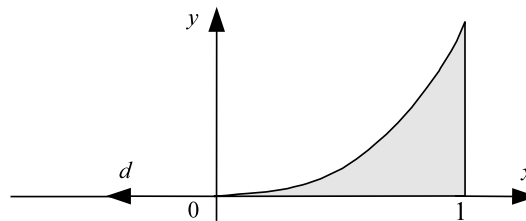


FIGURE 9.1 – Contraintes non qualifiées en $(0, 0)$.

$$\nabla g_1(x, y) = \begin{pmatrix} -3x \\ 1 \end{pmatrix}, \quad \nabla g_1(0, 0) = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{et} \quad \nabla g_3(0, 0) = \begin{pmatrix} 0 \\ -1 \end{pmatrix}.$$

La direction $d = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$ vérifie $d^t \nabla g_1(0, 0) = 0$ et $d^t \nabla g_3(0, 0) = 0$, la direction d appartient à $B(0, 0)$. Or, la seule direction admissible en $(0, 0)$ est la direction $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ (on a donc $d \in B(0, 0) \setminus A(0, 0)$). La direction d n'est pas limite d'une suite de directions admissibles : les contraintes ne sont pas qualifiées en $(0, 0)$.

On établit enfin le théorème suivant :

Théorème 9.11. On suppose que le problème admet un minimum local en un point x^* où les contraintes sont qualifiées. Alors, si on a $d \in B(x^*)$:

$$d^t \nabla f(x^*) \geq 0$$

(par conséquent, aucune direction admissible en x^* n'est de descente¹).

1. Rappelons qu'une direction de descente est une direction d vérifiant $d^t \nabla f(x^*) < 0$ (voir la partie 8.7, page 150).

Preuve. Soit (d^k) une suite de directions admissibles tendant vers d et soit ϕ_k la fonction associée à d^k . Soit $s > 0$. Il vient :

$$f[\phi_k(s)] = f(x^*) + s(d^k)^t \nabla f(x^*) + s\varepsilon(s)$$

où $\varepsilon(s) \rightarrow 0$ quand $s \rightarrow 0$. Si s est assez petit : $f[\phi_k(s)] \geq f(x^*)$. On a alors : $s[(d^k)^t \nabla f(x^*) + \varepsilon(s)] \geq 0$ et donc $(d^k)^t \nabla f(x^*) + \varepsilon(s) \geq 0$. Par passage à la limite quand s tend vers 0, on obtient $(d^k)^t \nabla f(x^*) \geq 0$. Par passage à la limite quand k tend vers $+\infty$, on obtient $d^t \nabla f(x^*) \geq 0$. \diamond

9.2 Conditions de Lagrange

On s'intéresse ici au problème :

$$\begin{aligned} & \text{Minimiser } f(x) \\ & \text{avec } \begin{cases} \text{pour } j \in J, & h_j(x) = 0 \\ x \in \mathbb{R}^n \end{cases} \end{aligned}$$

où les fonctions f et h_j ($j \in J$) sont de classe C^1 . Les conditions de Lagrange, que donne le théorème suivant, fournissent des conditions nécessaires pour qu'un élément de \mathbb{R}^n soit un minimum local de (P) .

Théorème 9.12. (Conditions de Lagrange) Soit x^* un minimum local du problème. On suppose que les contraintes sont qualifiées en x^* . Alors il existe p nombres réels μ_j ($j \in J$) vérifiant $\nabla f(x^*) = \sum_{j \in J} \mu_j \nabla h_j(x^*)$.

Preuve. Notons E le sous-espace de \mathbb{R}^n engendré par les vecteurs $\nabla h_j(x^*)$ ($j \in J$) et E^\perp le sous-espace orthogonal à E . On a :

$$\nabla f(x^*) = y + z \text{ avec } y \in E \text{ et } z \in E^\perp.$$

Pour $j \in J$, $(-z)^t \nabla h_j(x^*) = 0$ puisque $-z$ appartient à E^\perp . Par conséquent, $-z$ appartient à $B(x^*)$; d’après le théorème 9.11. (applicable puisque les contraintes sont qualifiées), il vient : $(-z)^t \nabla f(x^*) \geq 0$. Or :

$$(-z)^t \nabla f(x^*) = (-z)^t y + (-z)^t z = (-z)^t z = -\|z\|^2.$$

La relation $-\|z\|^2 \geq 0$ donne $z = 0$ et donc $\nabla f(x^*) \in E$. D’où le théorème. \diamond

Le théorème 9.13., conséquence directe du théorème 9.3 (page 176), donne des hypothèses pour lesquelles les conditions de Lagrange sont suffisantes.

Théorème 9.13. Les conditions de Lagrange sont suffisantes lorsque f est convexe dans un voisinage de x^* et que les h_j ($j \in J$) sont affines.

9.3 Conditions de Karush, Kuhn et Tucker

On reprend le problème (P) initial :

$$\text{Minimiser } f(x) \text{ avec } \begin{cases} \text{pour } i \in I, & g_i(x) \leq 0 \\ \text{pour } j \in J, & h_j(x) = 0 \\ x \in \mathbb{R}^n \end{cases}$$

où les fonctions f , g_i ($i \in I$) et h_j ($j \in J$) sont supposées de classe C^1 . Les conditions suivantes, appelées *conditions de Karush, Kuhn et Tucker*², donnent des conditions nécessaires d’optimalité qui généralisent les conditions de Lagrange :

2. W. Karush, *Minima of Functions of Several Variables with Inequalities as Side Conditions*, master thesis, université de Chicago, 1939; H. W. Kuhn, A. W. Tucker, Non-linear programming, in J. Neyman (dir.), *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, 1951, 481-492. Ces conditions sont aussi appelées *conditions de Kuhn et Tucker*. En 1951, Kuhn et Tucker ne connaissaient pas les travaux de Karush, alors peu diffusés. Ce n’est qu’en 1974 qu’ils en prirent connaissance et proposèrent à Karush d’ajouter son nom aux leurs.

Théorème 9.14. (conditions de Karush, Kuhn et Tucker)

On suppose que les contraintes sont qualifiées en x^* et que x^* est un minimum local du problème ; alors il existe :

- $|I_0(x^*)|$ nombres réels positifs ou nuls λ_i pour $i \in I_0(x^*)$
- p nombres réels μ_j ($j \in J$)

$$\text{vérifiant } \nabla f(x^*) = \sum_{j \in J} \mu_j \nabla h_j(x^*) - \sum_{i \in I_0(x^*)} \lambda_i \nabla g_i(x^*).$$

Remarques.

1. Les nombres λ_i et μ_j sont aussi appelés *multiplicateurs de Lagrange*.
2. On constate que dans l'expression de $\nabla f(x^*)$, seules les contraintes saturées interviennent.

Preuve. Soit $d = (d_k)_{1 \leq k \leq n}$ vérifiant :

- pour tout $i \in I_0(x^*)$, $\sum_{k=1}^n \frac{\partial g_i}{\partial x_k}(x^*) d_k \leq 0$
- pour tout $j \in J$, $\sum_{k=1}^n \frac{\partial h_j}{\partial x_k}(x^*) d_k = 0$.

Cela signifie que d appartient à $B(x^*)$. D'après le théorème 9.11., on a $d^t \nabla f(x^*) \geq 0$, c'est-à-dire : $\sum_{k=1}^n \frac{\partial f}{\partial x_k}(x^*) d_k \geq 0$. On pose :

- pour tout $i \in I_0(x^*)$ et tout $k \in \{1, 2, \dots, n\}$, $a_{ik} = -\frac{\partial g_i}{\partial x_k}(x^*)$
- pour tout $j \in J$ et tout $k \in \{1, 2, \dots, n\}$, $b_{jk} = \frac{\partial h_j}{\partial x_k}(x^*)$
- pour tout $k \in \{1, 2, \dots, n\}$, $c_k = \frac{\partial f}{\partial x_k}(x^*)$.

Avec ces notations, le résultat ci-dessus se réécrit :
si on a

- pour tout $i \in I_0(x^*)$, $\sum_{k=1}^n a_{ik}d_k \geq 0$,
- pour tout $j \in J$, $\sum_{k=1}^n b_{jk}d_k = 0$,

alors on doit avoir $\sum_{k=1}^n c_kd_k \geq 0$.

Le théorème de Farkas (voir l'exercice 4 du chapitre 6, page 88) montre qu'il existe λ_i pour $i \in I_0(x^*)$ et μ_j pour $j \in J$ vérifiant :

- pour $k \in \{1, \dots, n\}$, $\sum_{i \in I_0(x^*)} a_{ik}\lambda_i + \sum_{j \in J} b_{jk}\mu_j = c_k$
- pour $i \in I_0(x^*)$, $\lambda_i \geq 0$.

La première ligne s'écrit :

$$\bullet \text{ pour } k \in \{1, \dots, n\}, \sum_{j \in J} \mu_j \frac{\partial h_j}{\partial x_k}(x^*) - \sum_{i \in I_0(x^*)} \lambda_i \frac{\partial g_i}{\partial x_k}(x^*) = \frac{\partial f}{\partial x_k}(x^*)$$

$$\text{ou enfin : } \sum_{j \in J} \mu_j \nabla h_j(x^*) - \sum_{i \in I_0(x^*)} \lambda_i \nabla g_i(x^*) = \nabla f(x^*).$$

Avec la positivité des λ_i , on obtient l'énoncé du théorème de Karush, Kuhn et Tucker. \diamond

Nous illustrons ci-dessous deux cas où il n'y a pas de contrainte d'égalité ; les conditions de Karush, Kuhn et Tucker expriment alors qu'il est nécessaire que $\nabla f(x^*)$ se décompose sur l'ensemble

$\{-\nabla g_i(x^*) \text{ pour } i \in I_0(x^*)\}$ avec des coefficients positifs ou nuls.

Illustrations des conditions de Karush, Kuhn et Tucker.

★ *Cas $n = 2, p = 0$ et une seule contrainte d'inégalité saturée*

On suppose qu'on est dans \mathbb{R}^2 ; on note x_1 et x_2 les coordonnées d'un point. On suppose que seule la contrainte $g(x_1, x_2) \leq 0$ est saturée en (x_1^*, x_2^*) : $g(x_1^*, x_2^*) = 0$. Le vecteur $-\nabla g(x_1^*, x_2^*)$ est perpendiculaire à la courbe d'équation $g(x_1, x_2) = 0$ et dirigé vers l'intérieur du domaine. Si le vecteur $\nabla f(x_1^*, x_2^*)$ fait un angle non nul avec $-\nabla g(x_1^*, x_2^*)$, on peut trouver une direction de descente pour f dirigée vers l'intérieur du domaine et le point (x_1^*, x_2^*) n'est donc pas un minimum local.

La figure 9.2 illustre ce cas ; on rappelle qu'une direction est de descente si elle fait un angle obtus avec $\nabla f(x_1^*, x_2^*)$ et qu'une direction admissible fait un angle aigu avec $-\nabla g(x_1^*, x_2^*)$. Seul le cas où $\nabla f(x_1^*, x_2^*)$ fait un angle nul avec $-\nabla g(x_1^*, x_2^*)$ est compatible avec l'optimalité locale de (x_1^*, x_2^*) .

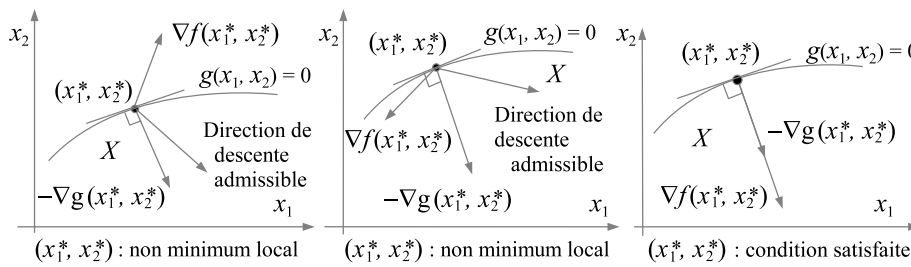


FIGURE 9.2 – Dans le plan, une seule contrainte saturée.

★ *Cas $n = 2, p = 0$ et deux contraintes d'inégalité saturées*

Sur la figure 9.3, on voit que pour qu'aucune direction de descente ne soit dirigée vers le domaine, il est nécessaire que le vecteur $\nabla f(x_1^*, x_2^*)$ se situe dans le secteur délimité par $-\nabla g_1(x_1^*, x_2^*)$ et

$-\nabla g_2(x_1^*, x_2^*)$. C'est le cas sur la figure.

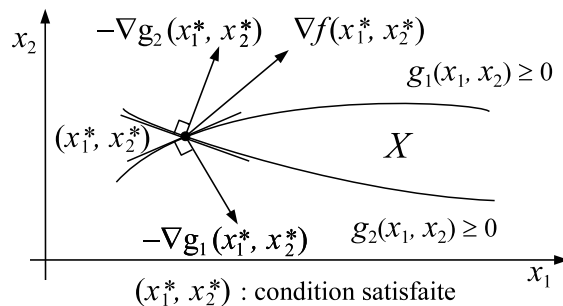


FIGURE 9.3 – Dans le plan, deux contraintes saturées.

Le théorème 9.3 donne des hypothèses pour lesquelles les conditions de Karush, Kuhn et Tucker sont suffisantes pour un minimum local.

Théorème 9.15. On suppose que les contraintes sont qualifiées en un point x^* . Les conditions de Karush, Kuhn et Tucker en x^* sont suffisantes pour avoir un minimum local s’il existe un voisinage de x^* dans lequel on a simultanément les fonctions f et g_i ($i \in I_0(x^*)$) convexes et les fonctions h_j ($j \in J$) affines.

Preuve. On suppose qu’il existe des nombres réels positifs ou nuls λ_i ($i \in I_0(x^*)$) et des nombres réels μ_j ($1 \leq j \leq p$) tels que :

$$\nabla f(x^*) = \sum_{j \in J} \mu_j \nabla h_j(x^*) - \sum_{i \in I_0(x^*)} \lambda_i \nabla g_i(x^*).$$

On considère une boule \mathcal{B} de centre x^* dans laquelle les fonctions f et g_i ($i \in I_0(x^*)$) sont convexes et les fonctions h_j ($j \in J$) affines. Soit $x \in \mathcal{B} \cap X$, on va montrer l’inégalité $f(x) \geq f(x^*)$, ce qui prouvera le théorème.

La convexité de f dans \mathcal{B} induit : $f(x) \geq f(x^*) + (x - x^*)^t \nabla f(x^*)$. En utilisant les conditions de Karush, Kuhn et Tucker :

$$f(x) \geq f(x^*) + \sum_{j \in J} \mu_j (x - x^*)^t \nabla h_j(x^*) - \sum_{i \in I_0(x^*)} \lambda_i (x - x^*)^t \nabla g_i(x^*).$$

Pour $j \in J$: $(x - x^*)^t \nabla h_j(x^*) = h_j(x) - h_j(x^*) = 0$.
Soit $i \in I_0(x^*)$; la fonction g_i étant convexe dans \mathcal{B} :

$$g_i(x) \geq g_i(x^*) + (x - x^*)^t \nabla g_i(x^*).$$

On a donc :

$$(x - x^*)^t \nabla g_i(x^*) \leq g_i(x) - g_i(x^*).$$

Or, on a $\lambda_i \geq 0$; de plus, par hypothèse, $g_i(x^*) = 0$ et $g_i(x) \leq 0$, d'où :

$$\lambda_i (x - x^*)^t \nabla g_i(x^*) \leq 0.$$

On obtient finalement $f(x) \geq f(x^*)$: f admet un minimum local en x^* . \diamond

9.4 Méthodes de descente

Dans cette partie 9.4, nous nous intéressons au problème suivant (sans perte de généralité, puisqu'une égalité peut se modéliser à l'aide de deux inégalités) :

$$\begin{aligned} & \text{Minimiser } f(x) \\ & \text{avec, pour } 1 \leq i \leq m, \quad g_i(x) \leq 0. \end{aligned}$$

Pour tenter de résoudre ce problème, on choisit un point de départ $x^0 \in X$ et on construit de façon itérative une suite x^k de

X vérifiant $f(x^{k+1}) < f(x^k)$, jusqu'à ce qu'on estime avoir obtenu une approximation satisfaisante.

À partir de x^k , on recherche une direction de descente d qui ne fasse pas sortir « immédiatement » de X . On cherche alors, en se déplaçant dans la direction d , un point x^{k+1} de X meilleur que x^k (par exemple, en minimisant $f(x^k + sd)$ pour $s > 0$, avec la contrainte que $x^k + sd$ appartienne à X , si on sait résoudre ce nouveau problème). On recommence à partir de x^{k+1} tant qu'un certain critère d'arrêt n'est pas vérifié.

Pour choisir d , on peut résoudre le problème :

$$\begin{aligned} & \text{Minimiser } d^t \nabla f(x^k) \\ & \text{avec } \begin{cases} d^t \nabla g_i(x^k) \leq 0 \text{ pour tout } i \text{ tel que } g_i(x^k) = 0 \\ \|d\| = 1. \end{cases} \end{aligned}$$

On obtient ainsi la direction de plus grande pente compatible avec les contraintes du problème; on appelle *méthode de plus grande pente* la méthode de descente qui effectue ce choix pour d . Si $d = -\nabla f(x^k)$ ne convient pas à cause des contraintes (on sortirait de X en se déplaçant selon la direction $-\nabla f(x^k)$), la solution du problème précédent sature au moins une contrainte $d^t \nabla g_i(x^k) \leq 0$: on se déplace donc tangentiellement à la frontière de X .

On norme le vecteur d de façon à avoir un minimum fini : en effet, s'il existe une direction d telle que $d^t \nabla f(x^k)$ soit négatif, on pourrait obtenir artificiellement une valeur aussi petite que l'on veut en considérant la direction αd où α est un réel positif; on peut en revanche remplacer la condition $\|d\| = 1$ par une condition imposant que d soit de norme majorée par une constante fixée. Le choix de la norme euclidienne (associée au produit scalaire que l'on souhaite minimiser) permet d'obtenir la direction qui maximise l'angle avec $\nabla f(x^k)$ et donc la direction de $B(x^k)$ de plus grande pente. Cette formulation présente cependant l'inconvénient de faire intervenir une racine carrée. Pour éviter cela, on

peut remplacer la condition $\|d\| = 1$ par la condition équivalente $\|d\|^2 = d^t d = 1$. On obtient alors des contraintes quadratiques. Pour obtenir un problème linéaire, on peut remplacer la norme euclidienne dans la contrainte $\|d\| = 1$ par la norme infinie (voir l’annexe ??, page ??), c’est-à-dire par la condition $-1 \leq d_i \leq 1$ ($1 \leq i \leq n$). Dans ce cas, la direction retenue ne sera pas *a priori* la direction de plus grande pente compatible avec les contraintes, mais on pourra appliquer les méthodes d’optimisation linéaire.

La méthode, telle qu’elle vient d’être exposée, peut rencontrer des difficultés. Considérons l’exemple représenté sur la figure 9.4. Tout déplacement dans la direction d fait sortir de X . Il faut alors une procédure de projection pour que x^{k+1} soit dans X , procédure qui peut être schématiquement représentée par la figure 9.5. Remarquons néanmoins que cette projection n’est pas utile dans le cas où les contraintes sont affines.

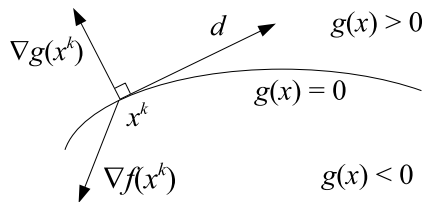


FIGURE 9.4 – Sortie du domaine.

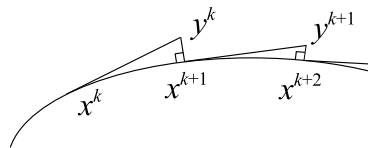


FIGURE 9.5 – Projection sur le domaine.

Une autre possibilité pour pallier cette difficulté consiste à remplacer les contraintes $d^t \nabla g_i(x^k) \leq 0$ par $d^t \nabla g_i(x^k) \leq -\varepsilon$, où ε est un paramètre positif. Ainsi, au lieu d’accepter une direction d qui parte tangentiellement à la surface de niveau $g_i(x) = 0$, ce qu’autorise la contrainte $d^t \nabla g_i(x^k) \leq 0$, on impose à d de « rentrer » dans le demi-espace d’équation $g_i(x) < 0$, au moins localement. La difficulté réside alors dans le choix de ε .

9.5 Cas des fonctions convexes

9.5.1 Généralités

On suppose dans toute cette partie 9.5 que le domaine de définition Ω de f est un ouvert convexe de \mathbb{R}^n et que :

- les fonctions g_i ($1 \leq i \leq m$) sont convexes sur Ω ,
- les fonctions h_j ($1 \leq j \leq p$) sont affines sur Ω ,
- l'intérieur du domaine réalisable X est non vide,
- la fonction f est convexe sur son domaine de définition Ω .

Remarques.

1. Si g est une fonction convexe sur Ω , l'ensemble des $x \in \Omega$ vérifiant $g(x) \leq 0$ est convexe.
2. Si h est une fonction affine sur Ω , l'ensemble des $x \in \Omega$ vérifiant $h(x) = 0$ est l'intersection de Ω avec un hyperplan et est donc convexe.
3. L'intersection de domaines convexes étant convexe, X est convexe.
4. D'après la proposition 9.7., les contraintes sont qualifiées en tout point de X .

Théorème 9.16. Si f est strictement convexe, le problème (P) admet au plus une solution optimale.

Preuve. Supposons qu'il existe dans X deux solutions optimales x et y ; on a donc $f(x) = f(y)$. Posons $z = \frac{x+y}{2}$. La convexité de X implique l'appartenance de z à X et la stricte convexité de f implique l'inégalité $f(z) < \frac{f(x)+f(y)}{2} = f(x)$, contradiction avec l'optimalité supposée de x . \diamond

Des théorèmes 9.1., page 164, et 9.16., on déduit :

Théorème 9.17. Si le domaine réalisable est borné et si f est strictement convexe, le problème (P) admet une unique solution optimale.

Des théorèmes 9.3., page 164, et 9.16., on déduit :

Théorème 9.18. Si f est strictement convexe et coercive, le problème (P) admet une unique solution optimale.

Théorème 9.19. Avec les hypothèses de cette partie, tout minimum local de (P) est global.

Preuve. Soit x^* un minimum local de (P) et soit $x \in X$. On définit une fonction ψ sur l'intervalle $[0, 1]$ par $\psi(s) = f[x^* + s(x - x^*)]$. On a $\psi(0) = f(x^*)$ et $\psi(1) = f(x)$. De plus, ψ est convexe puisque f l'est. Par ailleurs, on a $\psi'(0) = (x - x^*)^t \nabla f(x^*)$. La direction $d = x - x^*$ appartient à $A(x^*)$ et donc à $B(x^*)$. Le théorème 9.11., page 170, montre l'inégalité $\psi'(0) \geq 0$. La fonction ψ étant convexe sur $[0, 1]$, $\psi'(0) \geq 0$ implique $\psi(1) \geq \psi(0)$, c'est-à-dire : $f(x) \geq f(x^*)$. Par conséquent, x^* est bien un minimum global de (P) . \diamond

On peut maintenant donner des hypothèses pour que les conditions de Karush, Kuhn et Tucker soient suffisantes pour un minimum global en s'appuyant sur les théorèmes 9.3, 9.16. et 9.19.

Théorème 9.20. On suppose que les hypothèses fixées au début de la partie 9.5.1 sont vérifiées. Si les conditions de Karush, Kuhn et Tucker sont satisfaites en un point x^* , alors x^* est un minimum global de (P) . En outre, si f est strictement convexe, x^* est l'unique point où (P) atteint le minimum global.

Preuve. D’après le théorème 9.3, le problème (P) atteint un minimum local en x^* . D’après le théorème 9.19., x^* est un minimum global de (P) . Si, de plus, f est strictement convexe, le théorème 9.16. permet de conclure que x^* est l’unique minimum global de (P) . \diamond

9.5.2 Linéarisation : introduction

Dans les techniques de linéarisation, on considère une approximation de f par son développement de Taylor à l’ordre 1, et cela en tous les points d’une suite construite en utilisant cette linéarisation. Ceci conduit à un algorithme simple dont on va cependant constater les limites :

- $x^0 \leftarrow$ un point quelconque de X
- $k \leftarrow 0$
- répéter
 - ★ $x^{k+1} \leftarrow$ un point qui minimise $f(x^k) + (x - x^k)^t \nabla f(x^k)$ sur X
 - ★ $k \leftarrow k + 1$

jusqu’à ce qu’un test d’arrêt à préciser soit vérifié.

Remarques.

- 1) Le point x^{k+1} minimise aussi la fonction $x \mapsto x^t \nabla f(x^k)$ sur X puisque celle-ci ne diffère de la fonction $x \mapsto f(x^k) + (x - x^k)^t \nabla f(x^k)$ que par une constante.
- 2) Si le domaine X est un polyèdre, la détermination de x^{k+1} est un problème d’optimisation linéaire.

Appliquons cet algorithme au problème suivant dans \mathbb{R}^2 , noté (P_0) :

$$(P_0) \quad \begin{array}{l} \text{Minimiser } f(x_1, x_2) = (x_1 - 3)^2 + (x_2 - 5)^2 \\ \text{avec les contraintes : } \begin{cases} -2x_1 + x_2 \leq 0 \\ 2x_1 + x_2 - 20 \leq 0 \\ -2x_1 + 3x_2 - 4 \leq 0 \\ x_1 \geq 0, x_2 \geq 0. \end{cases} \end{array}$$

On remarque que l'on est bien dans le cadre général de la partie 9.5. La fonction objectif est constante sur des cercles centrés sur le point C de coordonnées $(3, 5)$. On peut donc anticiper sur le fait qu'elle est minimum pour le point du domaine le plus proche de C , c'est-à-dire le point $\left(\frac{49}{13}, \frac{50}{13}\right)$. On note X le domaine réalisable, grisé sur la figure 9.6. On a : $\nabla f(x_1, x_2) = \begin{pmatrix} 2x_1 - 6 \\ 2x_2 - 10 \end{pmatrix}$.

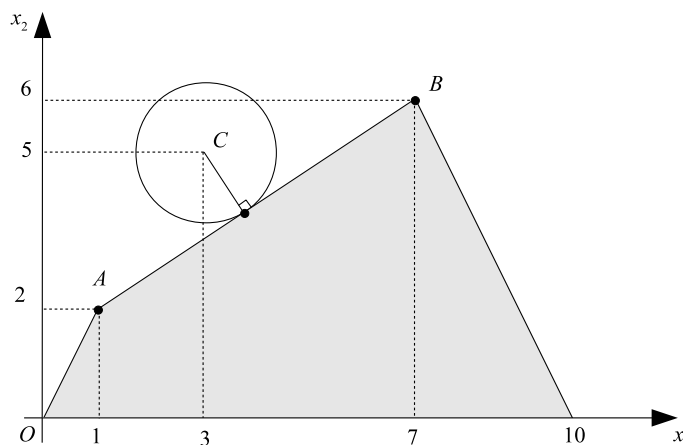


FIGURE 9.6 – Domaine réalisable X du problème (P_0) .

On démarre l'algorithme à partir de l'origine O , avec $\nabla f(0,0) = \begin{pmatrix} -6 \\ -10 \end{pmatrix}$. On cherche le minimum sur X de

$-6x_1 - 10x_2$. Les considérations développées dans le chapitre 5 montrent que le minimum est atteint en un des quatre sommets de X ; il est facile de montrer qu’il s’agit du point $B = (7, 6)$.

On recommence à partir du sommet B , avec $\nabla f(7, 6) = \begin{pmatrix} 8 \\ 2 \end{pmatrix}$.

On cherche le minimum de $8x_1 + 2x_2$ sur X . Il est atteint en un des quatre sommets de X ; il s’agit du point de départ $O = (0, 0)$. La poursuite de la méthode alternerait l’obtention de O et de B : la méthode ne converge donc pas.

9.5.3 Linéarisation : méthode de Frank et Wolfe

La méthode de Frank et Wolfe³ s’applique dans le cas où X est compact. Elle peut être décrite de la manière suivante :

- $x^0 \leftarrow$ un point quelconque de X
- $k \leftarrow 0$
- répéter
 - ★ $\tilde{x}^k \leftarrow$ un point qui minimise $x^t \nabla f(x^k)$ sur X
 - ★ $x^{k+1} \leftarrow$ un point qui minimise f sur le segment $[x^k, \tilde{x}^k]$
 - ★ $k \leftarrow k + 1$

jusqu’à ce qu’un test d’arrêt à préciser soit vérifié.

Remarque. Si X est un polyèdre convexe, grâce à la linéarité de la fonction $x \mapsto x^t \nabla f(x^k)$, on cherche \tilde{x}^k uniquement en un sommet de X .

Proposition 9.21. Si, dans la méthode de Frank et Wolfe, on a $x^{k+1} = x^k$, alors le problème admet un minimum global en x^k .

3. M. Frank, P. Wolfe, An algorithm for quadratic programming, *Naval Research Logistics Quarterly* 3, 1956, 95–110.

9.5. CAS DES FONCTIONS CONVEXES

185

Preuve. Soit $x \in X$. La convexité de la fonction f implique :

$$f(x) - f(x^k) \geq (x - x^k)^t \nabla f(x^k).$$

Le choix de \tilde{x}^k donne : $x^t \nabla f(x^k) \geq (\tilde{x}^k)^t \nabla f(x^k)$. D'où :

$$f(x) - f(x^k) \geq (\tilde{x}^k - x^k)^t \nabla f(x^k).$$

Posons par ailleurs, pour $s \in [0, 1]$: $\phi(s) = f(x^k + s(\tilde{x}^k - x^k))$.

Le minimum de f sur le segment $[x^k, \tilde{x}^k]$ est obtenu en x^{k+1} , c'est-à-dire en x^k . La fonction ϕ atteint donc son minimum pour $s = 0$, d'où $\phi'(0) \geq 0$. Or : $\phi'(0) = (\tilde{x}^k - x^k)^t \nabla f(x^k)$. On obtient : $f(x) - f(x^k) \geq 0$; x^k est donc un minimum global de f sur X . \diamond

Appliquons cette méthode au problème (P_0) précédent à partir de l'origine $O = (0, 0)$. Le déroulement de l'algorithme est illustré par la figure 9.7.

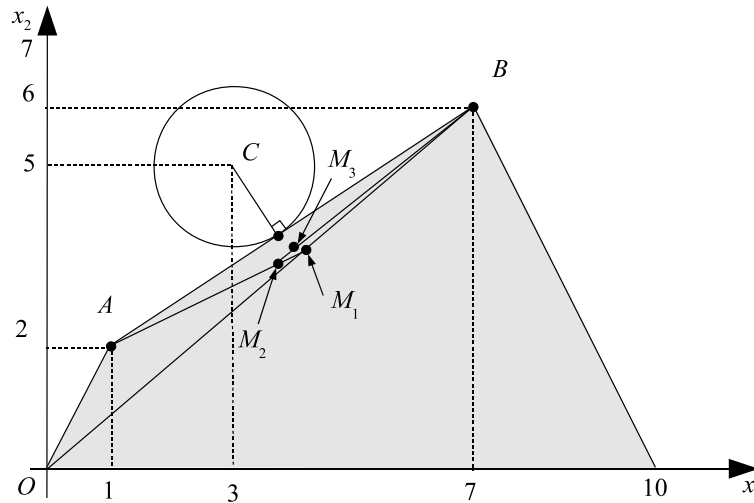


FIGURE 9.7 – Résolution de (P_0) par la méthode de Frank et Wolfe.

$$\text{On a : } \nabla f(x_1, x_2) = \begin{pmatrix} 2x_1 - 6 \\ 2x_2 - 10 \end{pmatrix}.$$

Étape 1.

On part du point $M_0 = (0, 0)$. On minimise $(x_1, x_2)\nabla f(0, 0) = -6x_1 - 10x_2$ sur X . Le point qui atteint ce minimum est $B = (7, 6)$.

On cherche le minimum de f sur le segment $[O, B]$. On paramètre ce segment par $x_1 = 7s, x_2 = 6s$ ($0 \leq s \leq 1$) et on pose $\phi_1(s) = f(7s, 6s)$, ce qui donne :

$$\phi_1(s) = (7s - 3)^2 + (6s - 5)^2 = 85s^2 - 102s + 34.$$

D'où : $\phi_1'(s) = 170s - 102$.

Le minimum de ϕ_1 est obtenu pour $s_1 = \frac{3}{5} = 0,6$.

La fonction f atteint donc son minimum sur le segment $[O, B]$ au point $M_1 = (7 \times 0,6 ; 6 \times 0,6) = (4,2 ; 3,6)$.

Étape 2.

On part du point $M_1 = (4,2 ; 3,6)$. On cherche le minimum sur X de $(x_1, x_2)\nabla f(4,2 ; 3,6) = 2,4x_1 - 2,8x_2$. Le point qui atteint ce minimum est le point $A = (1, 2)$.

On cherche le minimum de f sur le segment $[M_1, A]$.

On paramètre ce segment par : $\begin{cases} x_1 = s + 4,2(1 - s) \\ x_2 = 2s + 3,6(1 - s) \end{cases}$ ($0 \leq s \leq 1$)

ou encore : $\begin{cases} x_1 = -3,2s + 4,2 \\ x_2 = -1,6s + 3,6 \end{cases}$ ($0 \leq s \leq 1$).

On pose : $\phi_2(s) = f(-3,2s + 4,2 ; -1,6s + 3,6)$, ce qui donne :

$$\phi_2(s) = (-3,2s + 1,2)^2 + (-1,6s - 1,4)^2 = 12,8s^2 - 3,2s + 3,4.$$

D'où : $\phi_2'(s) = 25,6s - 3,2$.

Le minimum de ϕ_2 est obtenu pour $s = \frac{3,2}{25,6} = 0,125$.

La fonction f atteint donc son minimum sur le segment $[M_1, A]$ au point $M_2 = (-0,125 \times 3,2 + 4,2 ; -0,125 \times 1,6 + 3,6) = (3,8 ; 3,4)$.

Étape 3.

On part du point $M_2 = (3,8 ; 3,4)$. On cherche le minimum sur X de $(x_1, x_2)\nabla f(3,8 ; 3,4) = 1,6x_1 - 3,2x_2$. Le point qui atteint ce minimum est le point $B = (7, 6)$.

On cherche le minimum de f sur le segment $[M_2, B]$.

On paramètre ce segment par : $\begin{cases} x_1 = 3,8s + 7(1 - s) \\ x_2 = 3,4s + 6(1 - s) \end{cases} \quad (0 \leq s \leq 1)$

ou encore : $\begin{cases} x_1 = -3,2s + 7 \\ x_2 = -2,6s + 6 \end{cases} \quad (0 \leq s \leq 1)$.

On pose : $\phi_3(s) = f(-3,2s + 7 ; -2,6s + 6)$, ce qui donne :

$$\phi_3(s) = (-3,2s + 4)^2 + (-2,6s + 1)^2 = 17s^2 - 30,8s + 17.$$

D'où : $\phi_3'(s) = 34s - 30,8$.

Le minimum de ϕ_3 est obtenu pour $s = \frac{30,8}{34} \simeq 0,9059$.

La fonction f atteint donc son minimum sur le segment $[M_2, B]$ au point $M_3 = (-\frac{30,8}{34} \times 3,2 + 7 ; -\frac{30,8}{34} \times 2,6 + 6) \simeq (4,10 ; 3,64)$.

On peut continuer ainsi. Le théorème 9.22. qui suit montre que la suite des points M_k converge vers le minimum global du problème.

On aurait pu choisir un autre point de départ. Par exemple, recommençons l'algorithme en débutant au point $M'_0 = A$.

Étape 1.

On part du point $A = (1, 2)$. On minimise $(x_1, x_2)\nabla f(1, 2) = -4x_1 - 6x_2$ sur X . Le point qui atteint ce minimum est le point $B = (7, 6)$.

On cherche le minimum de f sur le segment $[A, B]$.

On paramètre ce segment par : $\begin{cases} x_1 = s + 7(1 - s) \\ x_2 = 2s + 6(1 - s) \end{cases} \quad (0 \leq s \leq 1)$

ou encore : $\begin{cases} x_1 = -6s + 7 \\ x_2 = -4s + 6 \end{cases} \quad (0 \leq s \leq 1).$

On pose : $\phi_1(s) = f(-6s + 7, -4s + 6)$, ce qui donne :

$$\phi_1(s) = f(-6s + 7, -4s + 6) = 52s^2 - 56s + 17.$$

D'où : $\phi_1'(s) = 104s - 56.$

Le minimum de ϕ_1 est obtenu pour $s = \frac{56}{104} = \frac{7}{13} \simeq 0,538.$

La fonction f atteint donc son minimum sur le segment $[A, B]$ au point $M'_1 = (-\frac{7}{13} \times 6 + 7 ; -\frac{7}{13} \times 4 + 6) = (\frac{49}{13}, \frac{50}{13}) \simeq (3,769 ; 3,846).$

Étape 2.

On part du point $M'_1 = (\frac{49}{13}, \frac{50}{13})$. On cherche le minimum sur X de :

$$(x_1, x_2) \nabla f(\frac{49}{13}, \frac{50}{13}) = \frac{20}{13}x_1 - \frac{30}{13}x_2 = \frac{10}{13}(2x_1 - 3x_2).$$

La fonction précédente est constante sur le segment $[A, B]$ (elle vaut $-\frac{40}{13}$) et son minimum sur X est obtenu sur tout ce segment. On choisit le point $A = (1, 2).$

On cherche le minimum de f sur le segment $[A, M'_1]$. Ce segment étant inclus dans le segment $[A, B]$, l'étape précédente montre que le minimum est de nouveau atteint en M'_1 . La proposition 9.21. montre que le problème admet un minimum global en M'_1 .

Théorème 9.22. Soit f une fonction définie sur un ouvert convexe Ω de \mathbb{R}^n à valeurs réelles, de classe C^1 ; on suppose que f est strictement convexe et que X est un polyèdre convexe compact de \mathbb{R}^n inclus dans Ω . La méthode de Frank et Wolfe appliqué au problème (P) de la minimisation de f sur X converge vers l’unique minimum global de (P) .

Pour prouver ce théorème, on établit d’abord le lemme suivant.

Lemme 9.23. Avec les hypothèses du théorème 9.22., soit $(x^k)_{k \in \mathbb{N}}$ une suite de points de X telle que la suite $f(x^k)_{k \in \mathbb{N}}$ soit décroissante. Supposons que la suite $(x^k)_{k \in \mathbb{N}}$ admette une sous-suite $(x^k)_{k \in K \subseteq \mathbb{N}}$ convergeant vers un minimum global de f . Alors la suite $(x^k)_{k \in \mathbb{N}}$ converge aussi vers ce minimum global.

Preuve du lemme. Notons x^* la limite de la sous-suite $(x^k)_{k \in K}$; supposons que la suite $(x^k)_{k \in \mathbb{N}}$ ne converge pas vers x^* . Alors, il existe $\varepsilon > 0$ tel que, pour tout $N \in \mathbb{N}$, il existe $k \geq N$ avec $\|x^k - x^*\| \geq \varepsilon$; on en déduit qu’il existe une sous-suite infinie $(x^k)_{k \in U \subseteq \mathbb{N}}$ de la suite $(x^k)_{k \in \mathbb{N}}$ vérifiant, pour tout $k \in U$, $\|x^k - x^*\| \geq \varepsilon$. Comme X est un compact, on peut extraire de la suite $(x^k)_{k \in U}$ une sous-suite $(x^k)_{k \in V \subseteq U}$ convergente; soit y^* la limite de cette sous-suite. Par passage à la limite, on a $\|y^* - x^*\| \geq \varepsilon$ et donc $y^* \neq x^*$.

La fonction f étant continue, les suites $(f(x^k))_{k \in K}$ et $(f(x^k))_{k \in V}$ convergent respectivement vers $f(x^*)$ et $f(y^*)$. Le point x^* donnant un minimum global de f , on a : $f(y^*) \geq f(x^*)$. Supposons que l’on ait $f(y^*) > f(x^*)$. Il existe $k_0 \in K$ tel que $f(x^{k_0}) < f(y^*)$. Soit $k \in V$ vérifiant $k \geq k_0$. La suite $f(x^k)$ étant décroissante, on a $f(x^k) \leq f(x^{k_0})$, ce qui implique $f(x^k) < f(y^*)$, contradiction avec le fait que la suite $(f(x^k))_{k \in V}$ converge en décroissant vers $f(y^*)$.

On a donc $f(y^*) = f(x^*)$, ce qui est impossible puisque la fonction f admet un unique minimum global sur X (cf. le théorème 9.17., page 181). Donc la suite $(x^k)_{k \in \mathbb{N}}$ converge vers x^* . \diamond

Preuve du théorème. Le théorème 9.17. montre déjà l’existence et l’unicité d’un minimum global. Si la suite construite par la méthode devient stationnaire après un nombre fini d’étapes, la proposition 9.21. montre que ce point stationnaire est le minimum global.

On suppose donc que ce n’est pas le cas et on note $(x^k)_{k \in \mathbb{N}}$ la suite construite par la méthode de Frank et Wolfe. Cette suite étant dans X qui est compact, elle admet une sous-suite convergente $(x^k)_{k \in K \subseteq \mathbb{N}}$; notons x^* la limite de cette suite. La suite $(\tilde{x}^k)_{k \in K}$ obtenue par la linéarisation prend chacune de ses valeurs en un des sommets du polyèdre; le nombre de sommets du polyèdre étant fini, on peut extraire de la suite $(x^k)_{k \in K}$ une sous-suite $(x^k)_{k \in H \subseteq K}$ telle que la suite $(\tilde{x}^k)_{k \in H}$ soit constante; on note \tilde{x} cette valeur constante.

La suite $(x^k)_{k \in H}$ converge vers x^* ; montrons que x^* constitue le minimum global du problème (P).

Soient s appartenant à $[0, 1]$ et k à H . La linéarisation en x^k prend son minimum en $\tilde{x}^k = \tilde{x}$; la méthode cherche le minimum de f sur le segment $[x^k, \tilde{x}^k] = [x^k, \tilde{x}]$ et donne le point x^{k+1} ; d’où $f(x^k + s(\tilde{x} - x^k)) \geq f(x^{k+1})$.

Soit $h(k)$ le plus petit des indices appartenant à H vérifiant $h \geq k + 1$; la suite $f(x^k)_{k \in \mathbb{N}}$ étant décroissante par construction des x^k , on a, pour tout $s \in [0, 1]$: $f(x^k + s(\tilde{x} - x^k)) \geq f(x^{h(k)+1})$.

En passant à la limite pour k dans H qui tend vers l’infini, on obtient :

$$f(x^* + s(\tilde{x} - x^*)) \geq f(x^*).$$

Écrivons la formule de Taylor de la fonction $s \mapsto f(x^* + s(\tilde{x} - x^*))$ au voisinage de 0, à l'ordre 1 :

$$f(x^* + s(\tilde{x} - x^*)) = f(x^*) + s(\tilde{x} - x^*)^t \nabla f(x^*) + s\varepsilon(s)$$

où $\varepsilon(s)$ tend vers 0 quand s tend vers 0.

En utilisant l'inégalité obtenue plus haut, il vient :

$$s(\tilde{x} - x^*)^t \nabla f(x^*) + s\varepsilon(s) \geq 0$$

ou encore, pour $s > 0$: $(\tilde{x} - x^*)^t \nabla f(x^*) + \varepsilon(s) \geq 0$.

En faisant tendre s vers 0, on a : $(\tilde{x} - x^*)^t \nabla f(x^*) \geq 0$, ce qui s'écrit :

$$\tilde{x}^t \nabla f(x^*) \geq (x^*)^t \nabla f(x^*). \quad (1)$$

Soit $k \in H$ et $x \in X$. Par construction de $\tilde{x}^k = \tilde{x}$, on a :

$$x^t \nabla f(x^k) \geq \tilde{x}^t \nabla f(x^k).$$

En passant à la limite quand k appartenant à H tend vers l'infini, on a :

$$x^t \nabla f(x^*) \geq \tilde{x}^t \nabla f(x^*). \quad (2)$$

En utilisant les inégalités (1) et (2), on obtient :

$$x^t \nabla f(x^*) \geq (x^*)^t \nabla f(x^*),$$

ou encore : $(x - x^*)^t \nabla f(x^*) \geq 0$.

La fonction f étant convexe, il vient : $(x - x^*)^t \nabla f(x^*) \leq f(x) - f(x^*)$.

On a donc maintenant : $f(x) - f(x^*) \geq 0$, ce qui montre que x^* est solution optimale du problème (P) .

Le lemme 9.23. montre que $(x^k)_{k \in \mathbb{N}}$ converge aussi vers x^* . \diamond

9.6 Exercices

Exercice 1

Énoncé. On s'intéresse au problème d'optimisation défini sur \mathbb{R}^2 de la façon suivante :

$$\begin{aligned} &\text{Minimiser } 2x_1^2 + x_2^4 \\ &\text{avec les contraintes } \begin{cases} x_1 \geq 1 \\ x_1 + ax_2 \geq a + 1 \end{cases} \end{aligned}$$

où a est un paramètre réel.

Q1. Pour quelles valeurs de a peut-on affirmer que le minimum global est atteint au point $(1, 1)$?

Q2. Résoudre le problème pour $a = 1/2$ à l'aide de la méthode de plus grande pente en partant du point $(1, 1)$.

Corrigé.

Q1. Mettons le problème sous la forme du cours et posons $f(x_1, x_2) = 2x_1^2 + x_2^4$, $g_1(x_1, x_2) = 1 - x_1$ et $g_2(x_1, x_2) = a + 1 - x_1 - ax_2$. Le problème s'écrit :

$$\begin{aligned} &\text{Minimiser } f(x_1, x_2) \\ &\text{avec les contraintes : } \begin{cases} g_1(x_1, x_2) \leq 0 \\ g_2(x_1, x_2) \leq 0. \end{cases} \end{aligned}$$

La matrice hessienne $\nabla^2 f(x_1, x_2) = \begin{pmatrix} 4 & 0 \\ 0 & 12x_2^2 \end{pmatrix}$ est positive :

la fonction f est convexe sur \mathbb{R}^2 .

Par ailleurs, une fonction affine est convexe (et d'ailleurs aussi concave) : g_1 et g_2 sont convexes. De plus, l'intérieur du domaine réalisable est non vide, il contient par exemple le point $(2, 1)$. La proposition 9.7., page 166, montre que les contraintes sont qualifiées en tout point de \mathbb{R}^2 . Par conséquent, d'après les théorèmes 9.14., page 173, et 9.20., page 181, il faut et il suffit que les conditions de Karush, Kuhn et Tucker soient vérifiées au point $(1, 1)$ pour que ce point soit un minimum global du problème. En ce point les deux contraintes sont saturées ; dire que les conditions de Karush, Kuhn et Tucker sont vérifiées revient à montrer qu'il

existe deux réels positifs ou nuls λ_1 et λ_2 vérifiant :

$$\nabla f(1, 1) = -\lambda_1 \nabla g_1(1, 1) - \lambda_2 \nabla g_2(1, 1).$$

Or, on a :

$$\nabla f(x_1, x_2) = \begin{pmatrix} 4x_1 \\ 4x_2^3 \end{pmatrix}, \nabla f(1, 1) = \begin{pmatrix} 4 \\ 4 \end{pmatrix},$$

$$\nabla g_1(1, 1) = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \nabla g_2(1, 1) = \begin{pmatrix} -1 \\ -a \end{pmatrix}.$$

Cherchons des coefficients λ_1 et λ_2 vérifiant : $\begin{pmatrix} 4 \\ 4 \end{pmatrix} = \lambda_1 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} 1 \\ a \end{pmatrix}$, ce qui s'écrit : $\begin{cases} \lambda_1 + \lambda_2 = 4 \\ a\lambda_2 = 4. \end{cases}$

Pour que ce système admette une solution, il faut et il suffit d'avoir $a \neq 0$ et alors :

$$\lambda_1 = 4 \left(1 - \frac{1}{a}\right), \lambda_2 = \frac{4}{a}.$$

Les conditions de Karush, Kuhn et Tucker sont vérifiées si et seulement si λ_1 et λ_2 sont positifs ou nuls, c'est-à-dire si et seulement si on a $a \geq 1$. Par conséquent, le point $(1, 1)$ est un minimum global du problème si et seulement si on a $a \geq 1$.

Q2. D'après ce qui précède, le minimum pour $a = 1/2$ n'est pas atteint au point $(1, 1)$. Appliquons la méthode de plus grande pente en partant du point $(1, 1)$: cherchons la direction admissible de plus grande pente pour f en ce point.

On voit graphiquement (voir la figure 9.8) que la direction $d = (1, -2)^t$ convient : c'est, parmi les directions admissibles, celle qui fait le plus grand angle avec $\nabla f(1, 1)$. Cherchons alors le

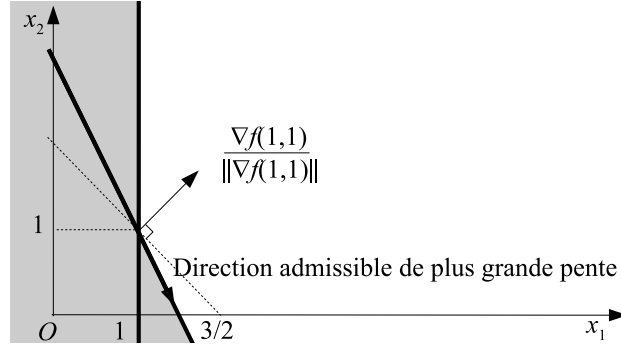


FIGURE 9.8 – Détermination graphique de la direction à suivre.

minimum de $\phi(s) = f((1, 1) + s(1, -2))$ pour s positif, en remarquant qu’ainsi on ne sort pas du domaine :

$$\phi(s) = 2(1 + s)^2 + (1 - 2s)^4.$$

D’où : $\phi'(s) = 4(1 + s) - 8(1 - 2s)^3$.

La fonction ϕ est convexe car f l’est. On cherche donc à annuler ϕ' , ce que l’on fait par une méthode de type dichotomique :

$\phi'(0) = -4 < 0$; $\phi'(1/2) = 6 > 0$; $\phi'(0,25) > 0$; $\phi'(0,125) > 0$; $\phi'(0,06) < 0$; $\phi'(0,09) < 0$; $\phi'(0,1) > 0$; $\phi'(0,095) > 0$; $\phi'(0,0925) > 0$; $\phi'(0,092) > 0$; $\phi'(0,091) < 0$; $\phi'(0,0915) > 0$; $\phi'(0,0913) < 0$; $\phi'(0,0914) < 0$; $\phi'(0,09145) > 0$; $\phi'(0,09142) > 0$; $\phi'(0,09141) > 0$:

$$0,09140 < s_{min} < 0,09141.$$

Le minimum de f dans la direction d est donc atteint au point $(1,0914 ; 0,8172)$.

Seule la contrainte g_2 est saturée en ce point. Regardons si les conditions de Karush, Kuhn et Tucker sont maintenant vérifiées :

$$\begin{aligned} \nabla f(1,0914 ; 0,8172) &\simeq \begin{pmatrix} 4,3656 \\ 2,1829 \end{pmatrix} \\ \nabla g_2(1,0914 ; 0,8172) &= \begin{pmatrix} -1 \\ -1/2 \end{pmatrix} \simeq \frac{-1}{4,3656} \begin{pmatrix} 4,3656 \\ 2,1829 \end{pmatrix}. \end{aligned}$$

Ici, $\nabla f(1,0914 ; 0,8172)$ et ∇g_2 sont colinéaires et de sens opposés. Les conditions de Karush, Kuhn et Tucker, nécessaires et suffisantes pour la minimalité globale, sont vérifiées. Le point $(1,0914 ; 0,8172)$ est solution globale du problème. Le minimum cherché vaut $f(1,0914 ; 0,8172) \simeq 2,828$.

Exercice 2

Énoncé. Soit α un paramètre réel de signe quelconque. On considère le problème (P_α) suivant :

$$\begin{aligned} &\text{Minimiser } f_\alpha(x, y) = x^2 + y^2 + xy + \alpha x \\ &\text{avec les contraintes } \begin{cases} x + y \geq 1 \\ x \geq 0. \end{cases} \end{aligned}$$

Q1. Indiquer, en fonction de α , les points du domaine réalisable où les contraintes sont qualifiées.

Q2. Montrer que, pour tout α , tout minimum local est global. On ne fera donc pas la distinction dans cet exercice.

Q3. En appliquant les conditions de Karush, Kuhn et Tucker, déterminer en fonction de α les coordonnées du point où f_α atteint son minimum sur le domaine considéré.

Q4. On considère maintenant le problème (P_1) obtenu pour $\alpha = 1$. Retrouver le résultat de la question précédente en appliquant la méthode de plus grande pente à pas optimal à partir du point $(1, 0)$. On fera un dessin représentant clairement la situation et sur lequel on s'appuiera pour justifier les directions suivies ou, à la fin, l'arrêt de la méthode.

Corrigé.

Q1. Commençons par écrire le problème pour le mettre sous la forme étudiée dans ce chapitre. Pour cela, posons $g_1(x, y) = 1 -$

$x - y$ et $g_2(x, y) = -x$. Le problème s’écrit :

minimiser f_α avec les contraintes $g_1(x, y) \leq 0$ et $g_2(x, y) \leq 0$.

Les fonctions g_1 et g_2 sont affines donc convexes. De plus, le domaine réalisable possède un intérieur strict non vide (celui-ci contient par exemple le point $(1, 1)$). La proposition 9.7., page 166, permet alors d’affirmer que les contraintes sont qualifiées en tout point, indépendamment des valeurs de α .

Q2. On a : $\nabla f_\alpha(x, y) = \begin{pmatrix} 2x + y + \alpha \\ x + 2y \end{pmatrix}$ et $\nabla^2 f_\alpha(x, y) = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$.

Le produit des valeurs propres de $\nabla^2 f_\alpha(x, y)$ est égal à son déterminant, c’est-à-dire 3 : les valeurs propres sont non nulles et de même signe. La somme des valeurs propres de $\nabla^2 f_\alpha(x, y)$ est égale à sa trace, c’est-à-dire 4 ; les deux valeurs propres sont strictement positives. La fonction f_α est donc strictement convexe. Le théorème 9.19., page 181, et la question 1 permettent de conclure que tout minimum local est global.

Q3. Le domaine réalisable est représenté par la figure 9.9.

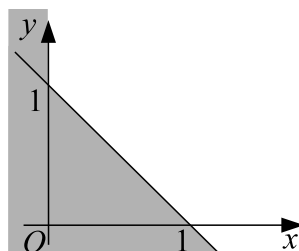


FIGURE 9.9 – Domaine réalisable pour le problème (P_α) .

La fonction f_α est convexe, les contraintes sont affines donc convexes et les contraintes sont qualifiées en tout point ; les conditions de Karush, Kuhn et Tucker sont donc nécessaires et suffi-

santes pour avoir un minimum (voir les théorèmes 9.14., page 173, et 9.20., page 181).

- Cherchons si le minimum peut se trouver à l’intérieur du domaine. Quand aucune contrainte n’est saturée, les conditions de Karush, Kuhn et Tucker se traduisent par l’annulation du gradient.

Le gradient s’annule si et seulement si on a $\begin{cases} 2x + y + \alpha = 0 \\ x + 2y = 0 \end{cases}$,

autrement dit si on a $x = -2\alpha/3$, $y = \alpha/3$. Ce point appartient à l’intérieur du domaine réalisable si et seulement si on a $2\alpha/3 < 0$ et $-2\alpha/3 + \alpha/3 > 1$, c’est-à-dire pour $\alpha < -3$.

- Cherchons maintenant quand les conditions de Karush, Kuhn et Tucker sont satisfaites en un point du bord d’équation $\begin{cases} x + y = 1 \\ x > 0. \end{cases}$

Seule la contrainte liée à g_1 étant saturée, les conditions de Karush, Kuhn et Tucker s’écrivent : $\nabla f_\alpha(x, y) = -\lambda_1 \nabla g_1(x, y)$, où λ_1 est un coefficient positif ou nul. Pour $x + y = 1$, on a $\nabla f_\alpha(x, y) = \begin{pmatrix} 2 - y + \alpha \\ y + 1 \end{pmatrix}$ et $\nabla g_1(x, y) = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$.

On doit donc avoir :
$$\begin{cases} \lambda_1 = 2 - y + \alpha \\ \lambda_1 = y + 1 \geq 0 \\ x = 1 - y \\ x > 0 \end{cases}, \text{ ou encore}$$

$$\begin{cases} x = \frac{1 - \alpha}{2} \\ y = \frac{\alpha + 1}{2} \\ \frac{\alpha + 1}{2} + 1 \geq 0 \\ \alpha < 1 \end{cases}$$

et donc
$$\begin{cases} x = \frac{1 - \alpha}{2} \\ y = \frac{\alpha + 1}{2} \\ -3 \leq \alpha < 1. \end{cases}$$

- Considérons désormais l'autre bord : $\begin{cases} x + y > 1 \\ x = 0. \end{cases}$

Seule la contrainte liée à g_2 étant saturée, les conditions de Karush, Kuhn et Tucker s'écrivent : $\nabla f_\alpha(x, y) = -\lambda_2 \nabla g_2(x, y)$, où λ_2 est un coefficient positif ou nul.

Pour $x = 0$, on a $\nabla f_\alpha(x, y) = \begin{pmatrix} y + \alpha \\ 2y \end{pmatrix}$ et $\nabla g_2(x, y) = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$. On doit donc avoir $y = 0$: le point obtenu est l'origine, qui n'est pas réalisable. Quelle que soit la valeur de α , il n'y a donc pas de solution optimale sur ce bord.

- Regardons enfin à quelle condition sur α le minimum est au point $(0, 1)$ où les deux contraintes sont saturées. Les conditions de Karush, Kuhn et Tucker s'écrivent : $\nabla f_\alpha(x, y) = -\lambda_1 \nabla g_1(x, y) - \lambda_2 \nabla g_2(x, y)$, où λ_1 et λ_2 sont des coefficients

positifs ou nuls. En ce point, on a : $\nabla f_\alpha(x, y) = \begin{pmatrix} 1 + \alpha \\ 2 \end{pmatrix}$,
 $\nabla g_1(x, y) = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$ et $\nabla g_2(x, y) = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$. On calcule
 λ_1 et λ_2 par : $\begin{cases} 1 + \alpha = \lambda_1 + \lambda_2 \\ 2 = \lambda_1 \end{cases}$, ce qui donne $\lambda_1 = 2$,
 $\lambda_2 = \alpha - 1$; les conditions de Karush, Kuhn et Tucker sont
 vérifiées si et seulement si λ_1 et λ_2 sont positifs ou nuls, c'est-
 à-dire pour $\alpha \geq 1$.

- Conclusion : on a ainsi déterminé le minimum global du problème pour toutes les valeurs de α :
 - ★ pour $\alpha < -3$, le minimum est en $x = -2\alpha/3$, $y = \alpha/3$, à l'intérieur du domaine ;
 - ★ pour $-3 \leq \alpha < 1$, le minimum est en $x = (1 - \alpha)/2$, $y = (1 + \alpha)/2$, sur un des deux bords ;
 - ★ pour $\alpha \geq 1$, le minimum est en $(0, 1)$.

Q4. Au point $(1, 0)$, le gradient de f_1 vaut $(3, 1)^t$. Les directions de descente étant celles faisant un angle supérieur à $\pi/2$ avec le gradient de f_1 , les directions admissibles et de descente sont celles appartenant au secteur marqué par un arc de cercle représenté sur la figure 9.10.

Parmi ces directions admissibles et de descente, la direction de plus grande pente est celle qui s'éloigne (angulairement) le plus de $\nabla f_1(1, 0)$, c'est-à-dire la direction d qui suit la droite d'équation $g_1(x, y) = 0$ en remontant vers le point $(0, 1)$ (voir figure 9.10). On se déplace donc dans la direction et le sens donnés par le vecteur $(-1, 1)^t$. Le nouveau point cherché est de la forme $(1, 0)^t + s(-1, 1)^t = (1 - s, s)^t$ avec $s \geq 0$. Définissons la

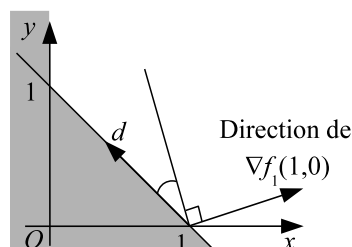


FIGURE 9.10 – Direction de $\nabla f_1(1,0)$ et direction d .

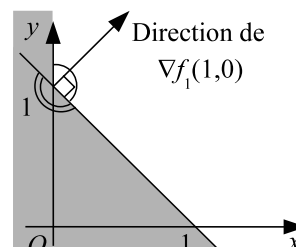


FIGURE 9.11 – Direction de $\nabla f_1(0,1)$.

fonction γ d’une variable s par $\gamma(s) = f_1(1-s, s) = s^2 - 2s + 2$. Comme $\gamma'(s)$ vaut $2s-2$, le minimum de γ est obtenu pour $s = 1$. On atteint alors le point $(0, 1)$, qui appartient bien au domaine réalisable.

Au point $(0, 1)$, le gradient de f_1 vaut désormais $(2, 2)^t$ (voir la figure 9.11). Aucune direction de descente n’est admissible. En effet, les directions admissibles sont celles appartenant au secteur marqué d’un seul petit arc dans la figure 9.11, bords inclus, alors que les directions de descente sont les directions faisant avec $\nabla f_1(0, 1)$ un angle supérieur à $\pi/2$, c’est-à-dire les directions du secteur marqué de deux petits arcs, bord exclu. On a atteint le minimum cherché : le point $(0, 1)$. Il correspond bien au résultat obtenu à la question précédente.

Annexe A

Rappels sur les graphes

A.1 Graphes non orientés

A.1.1 Notions fondamentales

Définition A.1. Un *graphe fini, non orienté, simple* $\mathcal{G} = (S, A)$ est défini par deux ensembles finis : l’ensemble S de ses *sommets* et l’ensemble A de ses *arêtes*, un élément a de A étant défini par une paire (non ordonnée) $\{s, t\}$ ($= \{t, s\}$) de sommets distincts s et t de S ; une arête donnée n’apparaît pas plusieurs fois dans \mathcal{G} .

Dans toute la partie A.1, les définitions sont relatives à un graphe non orienté (fini et simple) $\mathcal{G} = (S, A)$.

Définition A.2. Si le graphe \mathcal{G} possède une arête $a = \{s, t\}$, on dit que l’arête a est *incidente* à s et t , que s et t sont les *extrémités* de a , que s et t sont *adjacents*, que s est un *voisin* de t (et réciproquement). On note $V(s)$ et on appelle *voisinage* de s l’ensemble des voisins de s :

$$V(s) = \{t \in S \text{ avec } \{s, t\} \in A\}.$$

Les graphes non orientés peuvent être représentés par des figures. Des points (ou des cercles, des carrés, des croix...) cor-

respondent aux sommets, des segments de droite ou de courbe joignent les points représentant les extrémités des arêtes.

Définition A.3. L’*ordre* de \mathcal{G} est son nombre de sommets, c’est-à-dire le cardinal de S . La *taille* de \mathcal{G} est son nombre d’arêtes, c’est-à-dire le cardinal de A .

L’ordre de \mathcal{G} sera souvent noté n et sa taille m .

Proposition A.4. La taille maximum d’un graphe non orienté d’ordre n vaut $\frac{n(n-1)}{2}$.

Définition A.5. Soit \mathcal{G} un graphe non orienté et soit s un sommet de \mathcal{G} . Le *degré* de s est le nombre d’arêtes incidentes à s ; le degré de s est donc aussi le nombre de voisins de s . Le degré de s dans \mathcal{G} est souvent noté $d_{\mathcal{G}}(s)$ ou $\delta_{\mathcal{G}}(s)$, ou, quand il n’y a pas d’ambiguïté, $d(s)$ ou $\delta(s)$. Un sommet de degré nul est dit *isolé*.

Pour tout sommet s d’un graphe d’ordre n , le degré $d(s)$ de s vérifie les inégalités $0 \leq d(s) \leq n-1$.

Proposition A.6. Dans tout graphe non orienté, on a :

$$\sum_{s \in S} d(s) = 2m.$$

Définition A.7. Soit T une partie de S (qui peut être vide ou égale à S tout entier). Le graphe $\mathcal{H} = (T, B)$, où B est l’ensemble des arêtes de \mathcal{G} qui ont leurs deux extrémités dans T , s’appelle le *sous-graphe de \mathcal{G} engendré par T* . Autrement dit, à partir de \mathcal{G} , on obtient un sous-graphe en enlevant une partie des sommets et toutes les arêtes qui ont au moins une extrémité parmi les sommets enlevés.

Définition A.8. Soit B une partie de A (qui peut être vide ou égale à A tout entier). Le graphe $\mathcal{H} = (S, B)$ est appelé le *graphe partiel de \mathcal{G} engendré par B* . Autrement dit, à partir de \mathcal{G} , on obtient un graphe partiel en conservant tous les sommets de \mathcal{G} et en enlevant éventuellement des arêtes de \mathcal{G} .

Si, d’un sous-graphe de \mathcal{G} engendré par une partie T de sommets, on enlève une ou plusieurs arêtes qui ont leurs deux extrémités dans T (ce qui revient à combiner les deux définitions précédentes), on obtient un *sous-graphe partiel* de \mathcal{G} .

A.1.2 Chaînes et cycles

Définition A.9. Dans un graphe non orienté, on appelle *chaîne* une suite de sommets de la forme $s_0—s_1—s_2 \dots s_{p-1}—s_p$, avec $p \geq 0$, telle que, pour i compris entre 0 et $p-1$, $\{s_i, s_{i+1}\}$ est une arête.

Définition A.10. Soit \mathcal{G} un graphe non orienté et soit $s_0—s_1 \dots s_p$ une chaîne de \mathcal{G} . Les sommets s_0 et s_p s’appellent les *extrémités* de cette chaîne. On dit que la chaîne *joint* ou *relie* les sommets s_0 et s_p .

Définition A.11. Une *chaîne* est dite *élémentaire* si elle ne contient pas deux fois le même sommet ; autrement dit, une chaîne $s_0—s_1—s_2 \dots s_{p-1}—s_p$ est élémentaire si, pour i et j compris entre 0 et p , $i \neq j$ implique $s_i \neq s_j$.

Définition A.12. Dans un graphe non orienté, on appelle *cycle élémentaire* une suite de la forme $s_0—s_1—s_2 \dots s_{p-1}—s_p—s_0$, avec $p \geq 0$, telle que :

- pour i compris entre 0 et $p-1$, $\{s_i, s_{i+1}\}$ est une arête ;

- $\{s_p, s_0\}$ est une arête ;
- pour i et j compris entre 0 et p , $i \neq j$ implique $s_i \neq s_j$.

Un cycle est donc une chaîne fermée par l’arête reliant les deux extrémités de celle-ci.

A.1.3 Connexité

Définition A.13. Un graphe non orienté est dit *connexe* si, pour toute paire de sommets distincts, il existe une chaîne les joignant.

Définition A.14. Étant donné un graphe non orienté $\mathcal{G} = (S, A)$, on définit une relation d’équivalence sur S en posant que deux sommets s et t sont équivalents si et seulement s’ils sont reliés par une chaîne dans \mathcal{G} . Les sous-graphes engendrés par les classes d’équivalence de S pour cette relation s’appellent *composantes connexes* de \mathcal{G} .

Tout graphe se décompose de façon unique en ses composantes connexes. Un graphe est connexe si et seulement s’il n’a qu’une seule composante connexe.

A.1.4 Graphes particuliers

Cette partie introduit des familles de graphes particuliers : les arbres, les graphes complets et les graphes bipartis.

Définition A.15. On appelle *arbre* un graphe non orienté, simple, connexe et sans cycle.

Définition A.16. Un *graphe* non orienté simple est dit *complet* si toute paire de sommets distincts est une arête. On dit aussi qu’il s’agit d’une *clique*. Le graphe complet d’ordre n est noté K_n .

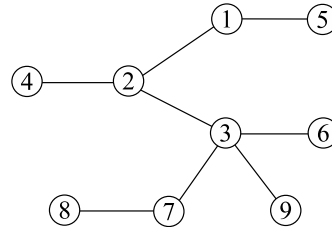


FIGURE A.1 – Un exemple d'arbre.

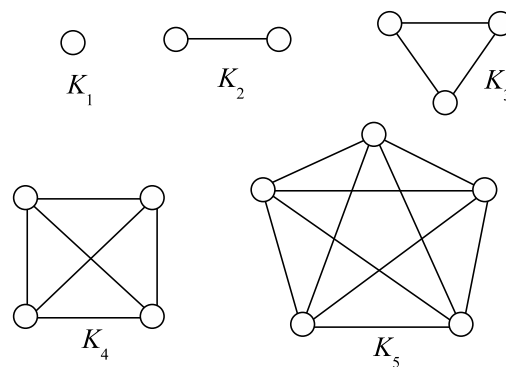


FIGURE A.2 – Graphes complets d'ordre 1, 2, 3, 4 et 5.

Le graphe complet K_n possède $\frac{n(n-1)}{2}$ arêtes et tous ses sommets sont de degré $n-1$. La figure A.2 représente les graphes complets d'ordre 1, 2, 3, 4 et 5.

Définition A.17. Un *graphe* $\mathcal{G} = (S, A)$ est dit *biparti* si on peut partitionner S en R et T (ce qui signifie que l'on a $R \neq \emptyset$, $T \neq \emptyset$, $R \cup T = S$, $R \cap T = \emptyset$) de sorte que toute arête ait une extrémité dans R et une extrémité dans T ; autrement dit, aucune arête n'a ses deux extrémités dans R ou ses deux extrémités dans T .

On trace quelquefois les graphes bipartis en mettant les sommets d'une partie sur une ligne et les sommets de l'autre partie sur une autre ligne. C'est ce qui est fait pour le graphe biparti représenté par la figure A.3.

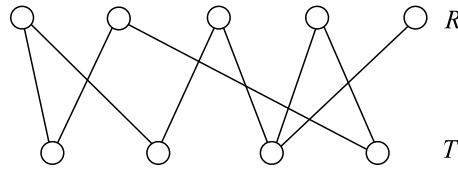


FIGURE A.3 – Un graphe biparti.

Notons p et q les cardinaux des deux parties R et T d'un graphe simple biparti \mathcal{G} . La taille de \mathcal{G} est au plus égale à $p \times q$. Si tout sommet de R est adjacent à tout sommet de T , on dit qu'il s'agit d'un *graphe biparti complet* et on le note $K_{p,q}$ (ou $K_{q,p}$). La figure A.4 représente $K_{4,3}$.

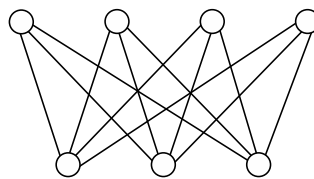


FIGURE A.4 – Le graphe $K_{4,3}$.

Proposition A.18. Un graphe est biparti si et seulement s'il n'a aucun cycle de longueur impaire.

A.2 Graphes orientés

A.2.1 Notions fondamentales

Définition A.19. Un *graphe fini orienté* $\mathcal{G} = (S, A)$ est défini par deux ensembles finis : l'ensemble S de ses *sommets* et l'ensemble A de ses *arcs*, un élément a de A étant défini par un couple (ordonné) de sommets s et t de S .

Dans toute la partie A.2, on considère un graphe orienté $\mathcal{G} = (S, A)$.

On notera que, dans un graphe orienté, l’arc (s, t) et l’arc (t, s) avec $s \neq t$ sont différents ; un graphe orienté peut donc posséder les arcs (s, t) et (t, s) simultanément.

Définition A.20. Soit $a = (s, t)$ un arc de \mathcal{G} ; on dit que :

- l’arc a est *incident* à s et t ;
- s est l’*origine* (ou aussi *extrémité initiale*) de l’arc a ;
- t est l’*extrémité* (ou aussi *extrémité finale* ou *terminale*) de l’arc a ;
- s et t sont *adjacents* ;
- s est un *prédécesseur* de t , ou aussi un *antécédent*, un *voisin intérieur* ou parfois un *père* (surtout quand s est le seul prédécesseur de t) de t ;
- t est un *successeur* de s , ou aussi un *voisin extérieur* ou un *fil* (surtout quand on considère que s est le père de t) de s .

On note $V^-(s)$ (respectivement $V^+(s)$) et on appelle *voisinage intérieur* (respectivement *voisinage extérieur*) l’ensemble des voisins intérieurs (respectivement voisins extérieurs) de s :

$$\begin{aligned} V^-(s) &= \{t \in S \text{ avec } (t, s) \in A\} ; \\ V^+(s) &= \{t \in S \text{ avec } (s, t) \in A\}. \end{aligned}$$

Les graphes orientés peuvent être représentés par des figures : des points (ou des cercles, ou des carrés...) correspondent aux sommets ; un arc $a = (s, t)$ est représenté par un segment de droite ou de courbe orienté allant de s à t .

Les figures A.5 et A.6 représentent deux graphes orientés. Les sommets de \mathcal{G}_1 ont pour noms 1, 2, 3, 4, 5, 6. Les arcs de \mathcal{G}_1 sont : $(1, 4)$, $(1, 5)$, $(2, 3)$, $(2, 4)$, $(3, 6)$, $(4, 1)$, $(5, 2)$, $(6, 2)$, $(6, 4)$. Le couple $(1, 2)$ n’est pas un arc de \mathcal{G}_1 ; le couple $(2, 5)$ non plus.

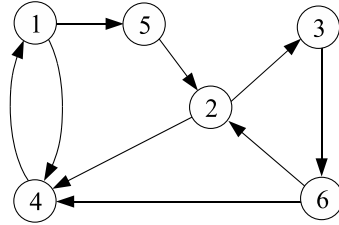


FIGURE A.5 – Le graphe \mathcal{G}_1 .

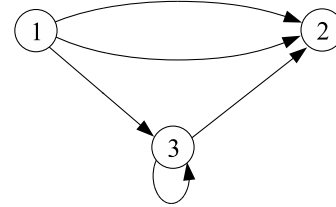


FIGURE A.6 – Le graphe \mathcal{G}_2 .

Définition A.21. On appelle *boucle* un arc de la forme (s, s) .

Dans le graphe \mathcal{G}_2 , l’arc $(3, 3)$ est une boucle.

Définition A.22. On dit qu’un *arc* (s, t) est *multiple* s’il figure plusieurs fois dans A .

Dans le graphe \mathcal{G}_2 , l’arc $(1, 2)$ est multiple (plus précisément il est double).

Définition A.23. On dit qu’un *graphe orienté* est *simple* s’il ne possède ni boucle ni arc multiple.

Le graphe \mathcal{G}_2 n’est pas simple (on dit qu’il s’agit d’un multigraphe) tandis que le graphe \mathcal{G}_1 l’est (la présence dans \mathcal{G}_1 des arcs $(1, 4)$ et $(4, 1)$ ne conduit pas à un arc multiple, l’orientation distinguant ces deux arcs).

Dans la suite, on supposera que les graphes orientés considérés sont simples.

Définition A.24. On dit qu’un *graphe orienté* est *symétrique* si, pour tout couple (s, t) de sommets, (s, t) est un arc si et seulement si (t, s) est un arc. On dit qu’un graphe orienté est *antisymétrique* ou *asymétrique* si, pour tout couple (s, t) de sommets, on a au plus un des deux arcs (s, t) ou (t, s) .

Remarque. Un graphe non orienté simple peut toujours être considéré comme un graphe orienté symétrique en remplaçant chacune de ses arêtes $\{s, t\}$ (avec $s \neq t$) par les arcs (s, t) et (t, s) .

Définition A.25. L’ordre d’un graphe orienté $\mathcal{G} = (S, A)$ est, comme pour un graphe non orienté, son nombre de sommets, c’est-à-dire le cardinal de S ; il sera souvent noté n . La *taille* de \mathcal{G} est son nombre d’arcs, c’est-à-dire le cardinal de A ; il sera souvent noté m .

Par exemple, le graphe \mathcal{G}_1 est d’ordre 6 et de taille 9.

Proposition A.26. La taille maximum d’un graphe orienté d’ordre n vaut $n(n - 1)$.

Définition A.27. Soit s un sommet de \mathcal{G} . Le *degré extérieur* de s est son nombre de successeurs, c’est-à-dire encore le nombre d’arcs d’origine s ; il est souvent noté $d_{\mathcal{G}}^+(s)$ ou, quand il n’y a pas d’ambiguïté, $d^+(s)$. Le *degré intérieur* de s est son nombre de prédécesseurs, c’est-à-dire encore le nombre d’arcs d’extrémité s ; il est souvent noté $d_{\mathcal{G}}^-(s)$ ou, quand il n’y a pas d’ambiguïté, $d^-(s)$.

Dans le graphe \mathcal{G}_1 , le sommet 4 est de degré extérieur 1 et de degré intérieur 3 : $d_{\mathcal{G}_1}^+(4) = 1$, $d_{\mathcal{G}_1}^-(4) = 3$.

Proposition A.28. Avec les notations précédentes, on a, dans tout graphe orienté : $\sum_{s \in S} d^+(s) = \sum_{s \in S} d^-(s) = m$.

Les deux définitions suivantes sont analogues, mais pour un graphe orienté, à celles données plus haut pour définir un sous-graphe et un graphe partiel d’un graphe non orienté. Comme précédemment, on peut combiner ces deux définitions pour obtenir un *sous-graphe partiel*.

Définition A.29. Soit T une partie de S (qui peut être vide ou égale à S tout entier). Le graphe $\mathcal{H} = (T, B)$, où B est l'ensemble des arcs de \mathcal{G} qui ont leur origine et leur extrémité dans T , s'appelle le *sous-graphe de \mathcal{G} engendré*, ou *induit*, par T . Autrement dit, à partir de \mathcal{G} , on obtient un sous-graphe en enlevant une partie des sommets et tous les arcs dont l'origine ou l'extrémité figurent parmi les sommets enlevés.

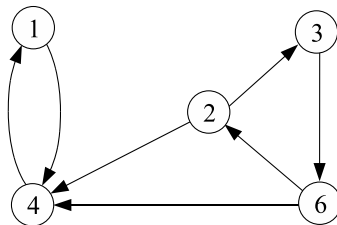


FIGURE A.7 – Le graphe \mathcal{G}_3 , sous-graphe de \mathcal{G}_1 .

Le graphe \mathcal{G}_3 de la figure A.7 est le sous-graphe de \mathcal{G}_1 engendré par l'ensemble de sommets $\{1, 2, 3, 4, 6\}$.

Définition A.30. Soit B une partie de A (qui peut être vide ou égale à A tout entier). Le graphe $\mathcal{H} = (S, B)$ est appelé le *graphe partiel de \mathcal{G} engendré*, ou *induit*, par B . Autrement dit, à partir de \mathcal{G} , on obtient un graphe partiel en conservant les sommets de \mathcal{G} et en enlevant éventuellement des arcs de \mathcal{G} .

Soit s_0 (respectivement a_0) un sommet (respectivement un arc) de \mathcal{G} . Comme pour les graphes non orientés, on notera $\mathcal{G} \setminus \{s_0\}$ (respectivement $\mathcal{G} \setminus \{a_0\}$) le sous-graphe (respectivement le graphe partiel) engendré par $S \setminus \{s_0\}$ (respectivement par $A \setminus \{a_0\}$).

La définition suivante complète le lien, commencé dans la remarque énoncée plus haut, page 209, entre graphes orientés et graphes non orientés.

Définition A.31. Soit $\mathcal{G} = (S, A)$ un graphe orienté. On appelle *graphe non orienté sous-jacent* à \mathcal{G} le graphe non orienté simple $\mathcal{G}' = (S, A')$ tel que, pour toute paire $\{s, t\}$ de sommets dans S , $\{s, t\}$ est dans A' si et seulement si (s, t) ou (t, s) sont dans A .

Autrement dit, on obtient le graphe sous-jacent d'un graphe orienté en enlevant l'orientation des arcs et en remplaçant alors d'éventuelles arêtes doubles par des arêtes simples.

Ceci permet notamment d'étendre la terminologie des graphes non orientés aux graphes orientés, en se référant alors aux graphes sous-jacents. On pourra par exemple parler de graphe orienté connexe ou de chaîne dans un graphe orienté, comme le précise la définition ci-dessous :

Définition A.32. Dans un graphe orienté \mathcal{G} , on appelle *chaîne* (respectivement *cycle*) de \mathcal{G} une chaîne (respectivement un cycle) du graphe non orienté sous-jacent à \mathcal{G} .

A.2.2 Chemins et circuits

Définition A.33. Dans un graphe orienté, on appelle *chemin* une suite de $p + 1$ ($p \geq 0$) sommets notée $s_0 \rightarrow s_1 \rightarrow \dots \rightarrow s_p$ telle que, pour i compris entre 0 et $p - 1$, (s_i, s_{i+1}) est un arc. La *longueur d'un chemin* est son nombre d'arcs. Un chemin est dit *élémentaire* s'il ne contient pas deux fois le même sommet (pour $0 \leq i < j \leq p$, $s_i \neq s_j$).

On accepte qu'un chemin puisse être réduit à un seul sommet ($p = 0$ avec les notations précédentes). La longueur d'un tel chemin est nulle.

Dans le graphe \mathcal{G}_5 (figure A.5), $1 \rightarrow 5 \rightarrow 2 \rightarrow 3 \rightarrow 6$ est un chemin élémentaire de longueur 4 tandis que $1 \rightarrow 5 \rightarrow 2 \rightarrow 3 \rightarrow 6 \rightarrow 2 \rightarrow 4$ est un chemin non élémentaire de longueur 6.

On pourra aussi utiliser cette notation pour représenter des chaînes dans des graphes orientés, surtout pour des graphes qui ne sont pas antisymétriques, afin de rappeler l’orientation des arcs constituant la chaîne. Ainsi, dans \mathcal{G}_1 , $5 \rightarrow 2 \leftarrow 6 \rightarrow 4$ est une chaîne élémentaire mais non un chemin.

Définition A.34. Dans un graphe orienté, on appelle *circuit* une suite de $p + 1$ ($p \geq 1$) sommets notée $s_0 \rightarrow s_1 \rightarrow \dots \rightarrow s_p \rightarrow s_0$, telle que, pour i compris entre 0 et $p - 1$, (s_i, s_{i+1}) est un arc ainsi que (s_p, s_0) . La *longueur d’un circuit* est son nombre d’arcs. Un circuit est dit *élémentaire* s’il ne contient pas deux fois le même sommet (pour $0 \leq i < j \leq p$, $s_i \neq s_j$).

Un circuit est toujours de longueur supérieure ou égale à 2 (avec les notations précédentes, une longueur égale à 2 correspond à $p = 1$).

Par exemple, dans le graphe \mathcal{G}_1 (figure A.5) :

- $1 \rightarrow 4 \rightarrow 1$ est un circuit de longueur 2 ;
- $1 \rightarrow 5 \rightarrow 2 \rightarrow 4 \rightarrow 1$ est un circuit élémentaire de longueur 4 ;
- $1 \rightarrow 5 \rightarrow 2 \rightarrow 3 \rightarrow 6 \rightarrow 2 \rightarrow 4 \rightarrow 1$ est un circuit non élémentaire de longueur 7 ;
- $2 \rightarrow 4 \leftarrow 6 \rightarrow 2$ est un cycle élémentaire (de longueur 3) mais non un circuit.

A.2.3 Forte connexité

Définition A.35. Un *graphe orienté* est dit *fortement connexe* si, pour toute paire de sommets distincts $\{s, t\}$, il existe un chemin de s à t et un chemin de t à s .

Le graphe \mathcal{G}_1 (figure A.5) est fortement connexe. Le graphe \mathcal{G}_3 de la figure A.7 n’est pas fortement connexe car il n’y a aucun chemin du sommet 1 au sommet 2.

Définition A.36. Étant donné un graphe $\mathcal{G} = (S, A)$, on définit une relation d’équivalence sur S en posant que deux sommets s et t sont équivalents si et seulement s’il existe dans \mathcal{G} un chemin de s à t et un chemin de t à s . Les sous-graphes engendrés par les classes d’équivalence de S pour cette relation s’appellent *composantes fortement connexes* de \mathcal{G} .

Par exemple, le graphe \mathcal{G}_3 de la figure A.7 possède deux composantes fortement connexes : le sous-graphe engendré par $\{1, 4\}$ et le sous-graphe engendré par $\{2, 3, 6\}$.

Un graphe est fortement connexe si et seulement s’il n’a qu’une seule composante fortement connexe (contenant alors tous les sommets du graphe). On remarquera aussi qu’il peut exister des arcs entre les composantes fortement connexes ; dans ce cas, tous les arcs entre deux composantes fortement connexes \mathcal{C} et \mathcal{C}' ont tous la même orientation (tous de \mathcal{C} vers \mathcal{C}' ou tous de \mathcal{C}' vers \mathcal{C}).

Bibliographie

Nous proposons ci-dessous une liste de quelques livres traitant divers aspects de l’analyse numérique ou de l’optimisation continue permettant d’approfondir les sujets abordés dans cet ouvrage. Certains couvrent un large domaine, d’autres au contraire sont plus spécialisés. Nous avons privilégié, quand cela était possible, des ouvrages récents écrits ou traduits en français. Nous n’avons cependant pas hésité à citer des manuels plus anciens ou en anglais, quand ceux-ci restent des références pour leurs thématiques.

- G. Allaire, *Analyse numérique et optimisation*, École polytechnique, 2005.
- L. Amodei, J.-P. Dedieu, *Analyse numérique matricielle*, Dunod, 2008.
- J. Bastien, J.-N. Martin, *Introduction à l’analyse numérique*, Dunod, 2003.
- M. Bergounioux, *Optimisation et contrôle des systèmes linéaires*, Dunod, 2001.
- M. Bierlaire, *Introduction à l’optimisation différentiable*, Presses polytechniques et universitaires romandes, 2006.
- F. Bonnans, *Optimisation continue. Cours et problèmes corrigés*, Dunod, 2006.
- S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2009.

- C. Brezinski, M. Redivo-Zaglia, *Méthodes numériques directes de l'algèbre matricielle*, Ellipses, 2005.
- G. Calafiore, L. El Ghaoui, *Optimization models*, Cambridge University Press, 2016.
- V. Chvátal, *Linear programming*, New York, W.H. Freeman and Company, 1983.
- P. Ciarlet, *Introduction à l'analyse numérique matricielle et à l'optimisation*, Dunod, 2007.
- P. Ciarlet, B. Miara, J.-M. Thomas, *Exercices d'analyse numérique matricielle et d'optimisation*, Dunod, 2001.
- M. Delfour, *Introduction à l'optimisation et au calcul semi-différentiel*, Dunod, 2012.
- F. Filbet, *Analyse numérique, algorithme et étude mathématique*, Dunod, 2013.
- S. Haddadi, *Programmation linéaire - Une approche mathématique et algorithmique*, Ellipses, 2021.
- J.-B. Hiriart-Urruty, *Optimisation et analyse convexe*, EDP Sciences, 2009.
- P. Lascaux, R. Théodor, *Analyse numérique matricielle appliquée à l'art de l'ingénieur*, Dunod, 2004.
- M. Minoux, *Programmation mathématique*, Lavoisier, 2008.
- C. Prins, M. Sevaux, *Programmation linéaire avec Excel*, Eyrolles, 2011.
- J. Rappaz, M. Picasso, *Introduction à l'analyse numérique*, Presses polytechniques et universitaires romandes, 2017.
- Roseaux (Groupe), *Exercices et problèmes résolus de recherche opérationnelle*, tomes 3, Dunod, 2003.
- R. Ruppli, *Programmation linéaire*, Ellipses, 2005.
- A. Ruszczyński, *Nonlinear Optimization*, Princeton University Press, 2011.

- R. Sioshansi, A. Conejo, *Optimization in engineering*, Springer, 2017.
- M. Terrenoire, D. Tounissoux, *Éléments de programmation mathématique*, Hermès, 1992.
- D. de Werra, *Éléments de programmation linéaire avec applications aux graphes*, Presses polytechniques romandes, 1990.
- D. de Werra, T. Liebling, J.-F. Hêche, *Recherche opérationnelle pour ingénieurs*, Presses polytechniques et universitaires romandes, 2003.

Index

- $A(.)$ (notation), 165
- algorithme
 - de Ford et Fulkerson, 93, 98, 112, 119, 125, 128
 - complexité, 106
 - convergence, 105, 121, 124
 - variante d’Edmonds et Karp, 107, 122
 - de marquage, 102
 - du simplexe, 43
 - critères de Dantzig, 54, 62, 64, 66
 - règle de Bland, 56, 64
- antécédent, 207
- arc, 206
 - incident, 207
 - multiple, 208
 - saturé, 94
- arête, 201
 - incidente, 201
- $B(.)$ (notation), 166
- base, 52
 - dégénérée, 54
- Bland
 - règle de, 56, 64
 - théorème de, 56
- boucle, 208
- capacité
 - d’un arc, 94
 - d’une coupe, 95
 - minimum, 93
 - résiduelle, 121
- certificat d’optimalité, 75
- certificat d’optimalité, 78
- chaîne, 203, 211
 - augmentante, 99
 - élémentaire, 203
- chemin, 211
 - élémentaire, 211
- circuit, 212
 - élémentaire, 212
- clique, 204
- compact, 143
- complexité, 15
- composante
 - connexe, 204
 - fortement connexe, 213
- conditionnement
 - d’un système linéaire, 16
 - d’une matrice, 17
- conditions

-
- de Karush, Kuhn et Tucker, 88, 172, 175, 176, 193
 - de Lagrange, 171
 - connexité, 204
 - forte, 212
 - contrainte
 - qualifiée, 166
 - saturée, 164
 - serrée, 164
 - convergence
 - linéaire, 152
 - quadratique, 152
 - superlinéaire, 152
 - d’ordre γ , 152
 - convexité
 - d’un polyèdre, 47
 - d’une fonction, 148
 - coupe, 95
 - capacité, 95
 - problème dual, 127
 - couplage, 110
 - cyclage, 54, 55, 64
 - cycle, 203, 211
 - Dantzig, 43
 - critères de, 54, 62, 64, 66
 - dégénérescence, 54, 64
 - degré, 202
 - extérieur, 209
 - intérieur, 209
 - dichotomie, 139, 141
 - dictionnaire, 48, 52
 - réalisable, 49, 58
 - direction
 - admissible, 165, 167, 170
 - de descente, 151, 152, 170
 - de plus grande pente, 152
 - domaine
 - admissible, 164
 - réalisable, 164
 - Edmonds et Karp (variante d’), 107, 122
 - erreur, 15
 - d’arrondi, 15
 - de troncature, 15
 - extrémité
 - d’un arc, 207
 - finale, 207
 - initiale, 207
 - terminale, 207
 - d’une arête, 201
 - factorisation LU, 27, 28
 - Farkas (théorème de), 174
 - fermé, 143
 - fil
 - d’un sommet, 207
 - flot, 94
 - (de valeur) maximum, 93
 - problème dual, 127

- valeur, 95
- flux d’un arc, 94
- fonction
 - affine, 166
 - coercive, 164
 - concave, 148
 - convexe, 148, 166, 180
 - objectif, 52
 - quadratique, 149
 - unimodale, 139, 141
- Ford et Fulkerson (algorithme de), 93, 98, 105–107, 112, 119, 121, 122, 124, 125, 128
- forme
 - quadratique, 149
 - standard, 45, 48, 51
- formule de Taylor, 144, 145
- forte connexité, 212
- Frank et Wolfe (méthode de), 184
- gradient, 144
 - méthode de, 152
- graphe, 201
 - antisymétrique, 208, 212
 - asymétrique, 208
 - biparti, 110, 205
 - complet, 206
 - complet, 204
 - connexe, 204
 - fortement connexe, 212
 - non orienté, 201, 206
 - sous-jacent, 211
 - orienté, 206
 - partiel, 210
 - simple, 208
 - symétrique, 208
- $I_0(.)$ (notation), 165
- inégalité
 - de Cauchy-Schwarz, 12
 - de Hölder, 12
 - de Minkowski, 12
- interpolation quadratique, 141
- Karush, Kuhn et Tucker
 - conditions de, 172, 175, 176, 193
 - théorème de, 172
- Lagrange
 - conditions de, 171
 - multiplicateur de, 173
- linéarisation, 182
- longueur
 - d’un chemin, 211
 - d’un circuit, 212
- matrice
 - adjointe, 9
 - de passage, 10

-
- définie positive, 146
 - diagonalisable, 10
 - équilibrage, 18
 - équivalente, 11
 - hermitienne, 10
 - hessienne, 145
 - normale, 10
 - orthogonale, 10
 - positive, 146
 - semblable, 10
 - symétrique, 10
 - trace, 14
 - unitaire, 10
 - maximum
 - global, 137
 - local, 137
 - méthode
 - à deux phases, 61, 67
 - de Cholesky, 30
 - de descente, 150, 151, 177
 - de direction admissible de plus grande pente, 192
 - de Frank et Wolfe, 184
 - de Gauss, 22
 - de Gauss-Jordan, 25
 - de gradient, 152
 - de la plus forte pente
 - à pas fixe, 153
 - à pas optimal, 153, 159
 - accélérée, 155
 - de Newton, 138, 156, 159
 - de plus grande pente, 178
 - de remontée, 21
 - par dichotomie, 139
 - par dichotomie sans dérivation, 141
 - par interpolation quadratique, 141
 - minimum
 - global, 137, 148, 164, 192
 - local, 137, 148, 151
 - multiplicateur de Lagrange, 173
 - Newton (méthode de), 138, 156, 159
 - norme, 12
 - équivalente, 13
 - matricielle, 13
 - euclidienne, 14
 - subordonnée, 13
 - vectorielle, 12
 - euclidienne, 12
 - infinie, 12
 - norme 1, 12
 - notations
 - $A(\cdot)$, 165
 - $B(\cdot)$, 166
 - $I_0(\cdot)$, 165
 - optimisation

- linéaire, 43, 44
- multidimensionnelle, 138, 142
- non linéaire
 - avec contraintes, 163
 - sans contrainte, 137
- unidimensionnelle, 138
- ordre d’un graphe, 202, 209
- origine d’un arc, 207
- ouvert, 143
- partie convexe, 148
- père
 - d’un sommet, 207
- pivot, 22
- polyèdre
 - convexe, 47
 - des contraintes, 46
- prédécesseur, 207
- prix implicite, 80
- problème
 - auxiliaire, 58, 67
 - borné, 52
 - d’affectation, 110, 127, 130
 - d’optimisation linéaire, 44
 - borné, 52
 - en nombres entiers, 51
 - infaisable, 52
 - non borné, 52
 - réalisable, 52, 67
 - de la coupe (de capacité) minimum, 93
 - du flot (de valeur) maximum, 93
- dual, 73
 - de la coupe minimum, 127
 - du flot maximum, 127
- dual-réalisable, 61
- non borné, 52
- non réalisable, 52
- primal, 74
- réalisable
 - borné, 52
 - non borné, 52
- produit
 - hermitien, 9
 - scalaire euclidien, 9
- programmation
 - linéaire, 43
- puits, 94
- qualification des contraintes, 166
- rayon spectral, 10
- règle
 - de Bland, 56, 64
- réseau, 94
- solution
 - basique, 49, 53

- hr/>
- dégénérée, 54
 - maximale, 137
 - minimale, 137
 - optimale, 52, 137
 - réalisable, 47, 52, 164
 - sommet, 201, 206
 - adjacent, 201, 207
 - isolé, 202
 - voisin, 201
 - source, 94
 - sous-graphe, 202, 210
 - partiel, 203, 209
 - spectre, 10
 - successeur, 207
 - systèmes linéaires, 21
 - taille
 - d’un graphe, 202, 209
 - Taylor (formule de), 144, 145
 - théorème
 - de Bland, 56
 - de Farkas, 88, 174
 - de Karush, Kuhn et Tucker, 172
 - de la dualité, 74
 - des écarts complémentaires, 78
 - du flot et de la coupe, 101
 - topologie, 143
 - valeur
 - propre, 10, 33
 - singulière, 11
 - valeur d’un flot, 95
 - valeur unitaire d’une ressource, 80
 - variable
 - d’écart, 48, 52
 - de base, 49, 52
 - de choix, 52
 - de décision, 52
 - entrante, 50, 53
 - hors-base, 49, 53
 - initiale, 52
 - principale, 52
 - sortante, 50, 54
 - versatile, 56
 - vecteur
 - adjoint, 9
 - propre, 10, 33
 - vitesse de convergence, 151
 - voisin, 201
 - extérieur, 207
 - intérieur, 207
 - voisinage, 201
 - extérieur, 207
 - intérieur, 207