

REPUBLIQUE ALGERIENNE DEMOCRATIQUE et POPULAIRE.  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique.

UNIVERSITE MOULOUD MAMMERY, TIZI-OUZOU  
Faculté des Sciences  
Département de Mathématiques



**MÉMOIRE DE MASTER**  
en  
**MATHÉMATIQUES**

Spécialité  
Méthodes et modèles de décision

Thème

**FLOT ET MULTIFLOT DANS L'OPTIMISATION DES  
RESEAUX  
DE TELECOMMUNICATION**

Présenté par

SIFAOUI Abdelhakim  
RAAB Lounes

Devant le jury d'examen composé de :

---

<i>M<sup>r</sup></i>	BACHIR SADI	M. de conférences A	Président
<i>M<sup>me</sup></i>	LESLOUS FADHILA	M. Assist A	Rapporteur
<i>M<sup>me</sup></i>	KHEFFACHE REZIKA	M. Assist A	Examinatrice

---

## *Remerciements*

Nous remercions avant tout ALLAH le tout puissant de nous avoir donné la force et le courage de réaliser ce travail.

Nous tenons à remercier notre promotrice, Mme LESLOUS, qui nous a encadrés pendant la réalisation de ce mémoire, et nous tenons à lui exprimer notre profonde reconnaissance pour son très bon encadrement, sa patience, sa gentillesse, sa disponibilité, nous la remercions vivement pour le temps qu'elle a consacrée à notre mémoire, sa confiance, de nous proposer un thème très intéressant.

Nos remerciements vont également aux étudiants de master RO, pour leurs grand soutien, leurs implication et leurs aides très fréquentes.

Sans oublier d'exprimer notre reconnaissance et gratitude à tous nos enseignants durant notre cursus au département Mathématiques, et aussi au primaire, au cem, et au lycée.

Nous tenons à remercier très sincèrement l'ensemble des membres du jury qui nous font le grand honneur d'avoir accepté de juger notre travail.

# Table des matières

<b>Introduction Générale</b>	<b>3</b>
<b>1 Optimisation des flots et de multiflots</b>	<b>5</b>
1.1 rappels et définitions . . . . .	5
1.2 Définition d'un flot . . . . .	7
1.2.1 Problème du flot compatible . . . . .	8
1.2.2 Problème du flot maximum . . . . .	8
1.3 Définition d'un multiflot . . . . .	9
1.3.1 Formulation sommets-arcs des problèmes de multi- flots . . . . .	11
1.3.2 Formulation arcs-chemins des problèmes de multiflot	12
1.3.3 Étude du problème du multiflot compatible . . . . .	15
1.3.4 Une méthode de décomposition par affectation de capacités . . . . .	22
<b>2 Les Réseaux</b>	<b>29</b>
2.1 Le réseau de Transport . . . . .	29
2.2 Le réseau de télécommunication . . . . .	30
2.3 Le problème de conception de réseaux . . . . .	31
2.4 État de l'art . . . . .	32
2.5 La notion de couche . . . . .	32
2.6 Les types de noeuds dans un réseau de télécommunication .	33
2.7 Construction de modèle . . . . .	34
2.7.1 Objectif . . . . .	34
2.7.2 Fonctions de coûts . . . . .	35
2.7.3 Formulation arcs-chemins . . . . .	36
2.7.4 Passage à la formulation capacité . . . . .	37
2.7.5 Introduction de couche . . . . .	40
2.7.6 La sécurisation . . . . .	41
2.7.7 Linéarisation du problème . . . . .	41

<i>TABLE DES MATIÈRES</i>	2
<b>Conclusion</b>	<b>44</b>
<b>Bibliographie</b>	<b>45</b>

# Introduction Générale

Aujourd'hui l'Internet s'est transformé en une architecture de communication globale supportant des services "temps-réel", tels que la voix sur IP ou la vidéo. Ces services ont des contraintes de qualité beaucoup plus strictes que les services traditionnels. Le développement attendu de ces services est susceptible de bouleverser radicalement les réseaux actuels, en engendrant de nouveaux usages, en attirant beaucoup plus d'utilisateurs et ainsi en accroissant de plusieurs ordres de grandeur les volumes de trafic qu'ils doivent supporter.

Toutes ces évolutions conduisent à une multiplication des services offerts par le réseau et à une croissance sans précédent du nombre d'utilisateurs et des volumes de trafics qu'il génèrent. Dans une société où l'information et la communication ont pris une telle importance, l'interruption des services offerts par le réseau, ou même une dégradation significative de la qualité de service, sont de moins en moins acceptables. Le respect des nouvelles exigences de qualité de service est ainsi un enjeu majeur.

Nous présentons nos travaux sur ces deux problématiques selon le plan suivant.

Le premier chapitre est consacré à la théorie des graphes. Dans un premier temps nous présentons les définitions sur les réseaux intervenant dans le routage et l'acheminement des flux en provenance des utilisateurs.

Ensuite on s'intéresse aux problèmes de (multi-)flots, à leur formulation, leur reformulation, leur résolution et les réductions que l'on peut apporter au graphe pour résoudre plus efficacement ces problèmes.

Nous proposons aussi une modélisation de flot et de multiflot dans les réseaux et nous précisons quelques propriétés des flux considérés dans ces réseaux. En particulier, nous présentons un algorithme pour vérifier l'existence d'un multiflot compatible.

Dans le second chapitre nous abordons la conception de réseau tolérant aux pannes permettant d'écouler un trafic donné.

Nous étudions tout d'abord le problème de la conception et du dimensionnement d'une topologie de communication résilience. Nous proposons un modèle de conception de réseaux intégrant les coûts et les contraintes des différents équipements (liens, routeurs, cartes). Ce modèle prend également en compte de nombreuses contraintes opérationnelles. Sachant que les capacités des liens prennent des valeurs dans un ensemble donné et que les fonctions de coûts à considérer ne sont non seulement pas linéaires, mais même pas continues.

Enfin nous présentons également les résultats de la comparaison entre les formulations sous forme d'un modèle mathématique, on déduit alors la formulation capacité du problème de dimensionnement de réseau sécurisé, qu'on peut représenter sous forme d'un programme linéaire.

# Chapitre 1

## Optimisation des flots et de multiflots

### Introduction

Dans ce chapitre on va donner quelques définitions fondamentales concernant les graphes, ainsi que des résultats sur les problèmes du flot et de multiflot, c'est à ce moment que les problèmes les plus pratiques font leur apparition : celui du flot maximum dans un réseau, celui de multiflot compatible.

### 1.1 Rappels et définitions

**Définition 1.1.** *Un graphe non-orienté  $G$  est défini par un ensemble de sommets ( paire de sommets ) on le not par :  $G = (X, E)$ .*

**Définition 1.2.** *Un graphe orienté (digraphe)  $G$  est défini par un ensemble d'arcs  $U$  ( paire de sommets orienté ) on le not par :  $G = (X, U)$ .*

**Définition 1.3.** *Un sommet est incident à une arête (un arc) si ce sommet forme une des extrémités de cette arête (cet arc) et vice-versa.*

**Définition 1.4.** *Deux sommets sont adjacents s'ils sont liés par une arête (un arc).*

**Définition 1.5.** *Deux arêtes (arcs) sont parallèles si elles (ils) correspondent à la même paire de sommets.*

**Définition 1.6.** *Un graphe simple est complet si toute paire de sommets définit une arête.*

**Définition 1.7.** *Un sous-graphe  $H$  d'un graphe  $G$  est défini par un sous-ensemble de nœuds  $N(H) \subseteq N(G)$  incidentes à ces sommets.*

**Définition 1.8.** *Un sous-graphe  $H$  forme un recouvrement (spanning) des sommets du graphe  $G$  si  $X(H) = X(G)$ .*

**Définition 1.9.** *Un chemin est un parcours simple, c-à-d un parcours dans lequel aucun sommet n'est répété : tous les sommets de la séquence  $i_0, i_1, \dots, i_n$  sont distincts.*

**Définition 1.10.** *Un cycle est un chemin  $i_0, i_1, \dots, i_n$  augmenté d'une arête  $(i_n, i_0)$ .*

**Définition 1.11.** *Un graphe est dit acyclique s'il ne contient pas de cycle orienté.*

**Définition 1.12.** *Deux sommets d'un graphe sont connectés s'il existe au moins un chemin entre ces deux sommets.*

**Définition 1.13.** *Un graphe est connexe s'il existe au moins un chemin entre chaque paire de sommets.*

**Définition 1.14.** *Un graphe est strictement connexe s'il existe au moins un chemin orienté entre chaque paire de sommets.*

**Définition 1.15.** *Si le nombre d'arcs qui va d'un sommet  $x_i$  à un sommet  $x_j$  ne peut jamais excéder un entier  $p$ , on dira qu'on a un  $P$ -graphe. Lorsque  $p = 1$ , le  $P$ -graphe devient un 1-graphe.*

*Exemples :*

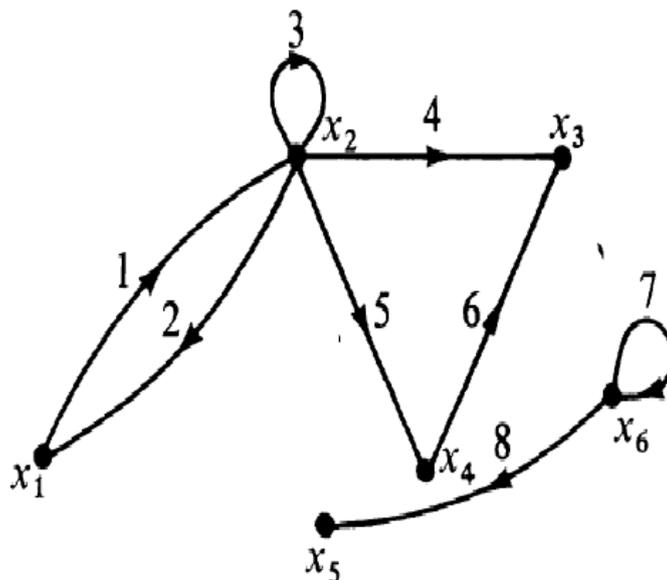


FIGURE 1.1 – Un 1-graphe d'ordre 6

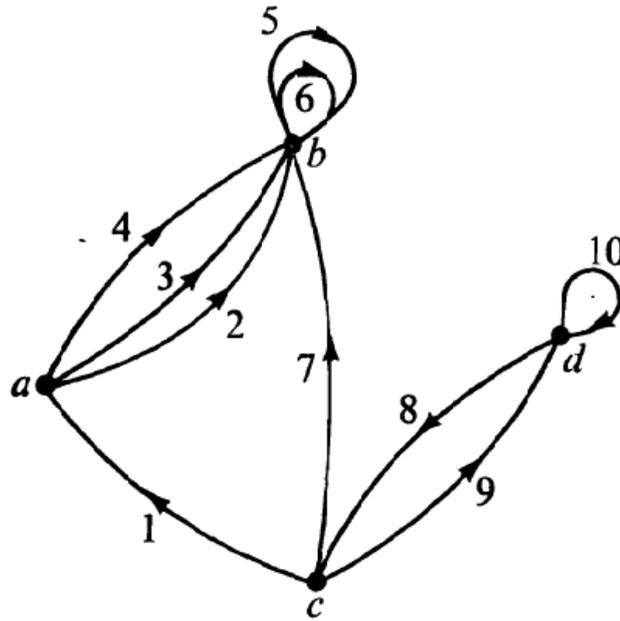


FIGURE 1.2 – Un 3-graphe d'ordre 4

**Définition 1.16.** *Graphe anti-symétrique.* Si pour tout couple

$(x, y) \in X \times X$ , on a

$$m_G^+(x, y) + m_G^-(x, y) \leq 1,$$

Le graphe  $G$  est dit anti-symétrique; dans le cas d'un 1-Graphe

$G = (X, U)$ , il est anti-symétrique si et seulement si on a :

$$(x, y) \in U \Rightarrow (y, x) \notin U,$$

Un 1-Graphe anti-ansymétrique considéré sans son orientation est donc un graphe simple, et vice versa.

**Définition 1.17.** On dit que l'on a effectué un routage des flots  $\varphi^1, \varphi^2, \dots, \varphi^K$  lorsqu'on a affecté à chaque flot  $k$  un chemin particulier  $Q_{\sigma(k)}^k$  pris parmi l'ensemble des flots dans  $p(k)$  chemins entre  $s_k$  et  $t_k$  ( $1 \leq \sigma(k) \leq p(k)$ ).

## 1.2 Définition d'un flot

Un flot représente l'acheminement d'un flux de matière depuis une source  $s$  vers une destination  $t$ . Le flot est ainsi décrit par la quantité de matière transitant sur chacun des arcs du réseau. Cette quantité doit être inférieure à la capacité de l'arc, qui limite ainsi le flux pouvant transiter par lui. De plus il n'est pas possible de stocker ou de produire de la matière aux sommet intermédiaires : un flot vérifie localement la loi de

conservation des flux en chaque sommet.

Considérons un graphe  $G$ , dont les arcs sont dénotés par  $1, 2, \dots, m$ ; considérons d'autre part dans  $\mathbb{R}$ , des nombres  $b_1, b_2, \dots, b_m, c_1, c_2, \dots, c_m$ , avec

$$-\infty \leq b_i \leq c_i \leq +\infty.$$

Un flot dans  $G$  est un vecteur  $\varphi = (\varphi_1, \varphi_2, \dots, \varphi_m) \in \mathbb{R}^m$ , tel que :

1. pour  $i = 1, 2, \dots, m$ ,  $\varphi_i$  est un nombre de  $\mathbb{R}$  appelé le flux dans l'arc  $i$ ; il sera assimilé à une quantité de matière parcourant l'arc  $i$  dans le sens de son orientation (si  $\varphi_i > 0$ ) ou en sens inverse ( $\varphi_i < 0$ );
2. pour tout sommet  $x$  du graphe, la somme des flux entrant dans  $x$  est égale à la somme algébrique des flux sortant de  $x$ ; c'est-à-dire :

$$\sum_{i \in \omega^-(x)} \varphi_i = \sum_{j \in \omega^+(x)} \varphi_j \quad (x \in X).$$

En d'autres termes, il y a conservation des flux en chaque sommet (loi de Kirchhoff).

### 1.2.1 Problème du flot compatible

Étant donné un graphe  $G$ , avec un intervalle  $[b_i, c_i]$  associé à tout arc  $i$  trouver un flot  $\varphi$  tel que  $b_i \leq \varphi_i \leq c_i$  ( $i = 1, 2, \dots, m$ ).

$c_i$  est appelé la capacité de l'arc  $i$ , et représente la quantité maximum de matière que peut utiliser l'arc  $i$  dans le sens de son orientation.

### 1.2.2 Problème du flot maximum

Étant donné un graphe  $G$ , avec un intervalle  $[b_i, c_i]$  associé à tout arc  $i$  trouver un flot  $\varphi$  tel que

1.  $b_i \leq \varphi_i \leq c_i$  ( $i = 1, 2, \dots, m$ );
2. le flux  $\varphi_1$  dans l'arc  $i = 1$  est aussi grand que possible.

remarquons que dans ces deux problèmes on pourra sans perdre de généralité supposer que  $G$  est un 1-graphe.

Le problème du flot maximum se rencontre le plus souvent dans le cas

particulier où  $G$  est un 1-graphe avec les conditions :

1.  $b_i = 0 \quad (i = 1, 2, \dots, m)$ ;
2.  $c_i > 0$  pour tout  $i$ , et  $c_i = +\infty$ ;
3. l'arc  $i=1$  est un arc  $(b, a)$ , reliant un point  $b$  appelé la sortie à un point  $a$  appelé l'entrée, ces deux points vérifiant :  
 $\omega^-(a) = \{1, 0, 0, \dots, 0\}$ ,  $\omega^+(b) = \{1, 0, 0, \dots, 0\}$ ,
4.  $G$  est un 1-graphe anti-symétrique.

En d'autres termes, il n'y a pas d'autres arcs que l'arc  $i = 1$  qui entre dans  $a$ , et il n'y a pas d'autres arcs que l'arc  $i = 1$  qui sort de  $b$ .

L'arc  $1 = (b, a)$ , qui ne sera en général pas dessiné sur les figures qui vont suivre, est appelé l'arc de retour, et n'a pour but que de maintenir aux sommets  $a$  et  $b$  la loi de conservation des flux. On cherche principalement le nombre maximum de matière que l'on peut envoyer de  $a$  vers  $b$  en suivant les arcs et en ne dépassant jamais la capacité des arcs.

**remarque 1.1.** *Les problèmes des flots sont des problèmes fondamentaux et très souvent utiles dès qu'il s'agit de trouver des chemins dans les graphes. Ils se départagent en deux catégories, le simple flot et le multiflot.*

### 1.3 Définition d'un multiflot

Un multiflot consiste à faire cohabiter plusieurs flots sur le réseau de sorte que la somme des flots passant sur un arc soit inférieure à la capacité. Et ce problème est utile dans un grand nombre d'application, notamment lorsqu'il est question de calculer des routes pour des entités qui sont en concurrence pour certaines ressources, ressources qui sont modélisées par des capacités sur les arcs du graphe support des routages possibles [18].

Exemple :

**Définition 1.18.** *Soit un graphe  $G = (X, U)$  décomposé par un ensemble  $X$  de  $n$  sommets et un ensemble  $U$  de  $m$  arcs. À chaque arc  $(i, j) \in U$  est associé un coût par unité de flot sur cet arc, Nous supposons que ce coût de flot sur un arc est linéaire par rapport au flot  $\varphi_{ij}$ . A chaque arc est aussi attribué une capacité  $c_{ij}$  qui décrit une borne supérieure du flot sur cet arc :  $\varphi_{ij} \leq c_{ij}$ .*

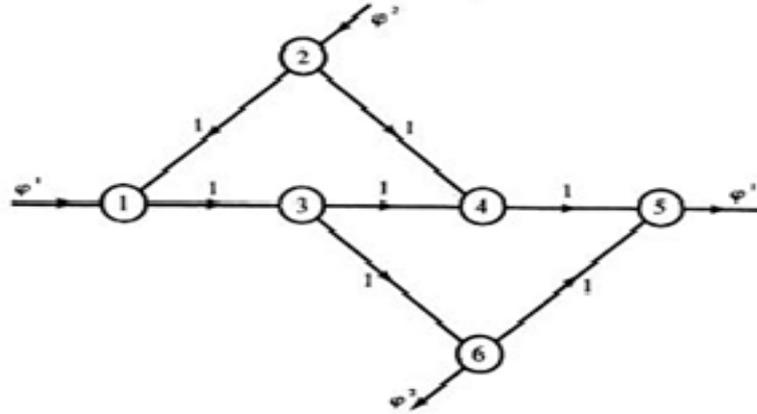


FIGURE 1.3 – Un réseau avec deux flots

De plus, on note par  $A$  la matrice d'incidence sommets-arcs, nous considérons aussi des flots de  $K$  type de produit distincts entre des paires de noeuds origine-destination :  $(s_1, t_1), (s_2, t_2), \dots, (s_K, t_K)$ . Le flot du type de produit  $K$  sur un arc  $(i, j) \in U$  s'écrit par  $\varphi_{ij}^K$ . Nous pouvons alors associer à chaque sommet  $i \in N$  un nombre entier  $b^K(i)$  représentant la source/demande d'un type de produit  $K$ . Si  $b^K(i) > 0$ , le sommet  $i$  est une source du type de produit  $K$ , Si  $b^K(i) < 0$ , il désigne une demande et pour  $b^K(i) = 0$ , la somme des flots des types de produit  $K$  entrant et sortant sont égaux. Il y a plusieurs critères d'optimisation d'un problème multiflots. L'un d'eux consiste à minimiser la somme des coûts et peut être formulé de la façon suivante :

$$\left\{ \begin{array}{l} \text{Min} \quad \sum_{(i,j) \in U} \sum_{k=1}^K c_{ij}^k \varphi_{ij}^k \\ \text{sc :} \\ \sum_{k=1}^K x_{ij}^k \leq u_{ij} \quad \text{pour chaque } (i, j) \in U \\ A \cdot \varphi^k = b^k \quad \text{pour } k = 1, 2, \dots, K \\ 0 \leq \varphi_{ij}^k \leq c_{ij}^k \quad \forall (i, j) \in U \text{ et } k = 1, 2, \dots, K \end{array} \right.$$

### 1.3.1 Formulation sommets-arcs des problèmes de multiflots

#### Problème du multiflot maximum(MM) sur un graphe avec bornes supérieures de capacité

À chaque arc  $u \in U$  de  $G$  on associe un nombre  $c_u > 0$  appelé la capacité de l'arc.

on cherche un multiflot  $\Psi = (\Psi_u)$  de valeur totale  $Z = z^1 + z^2 + \dots + z^K$  maximale sous les conditions de capacité :  $0 \leq \Psi_u \leq c_u (\forall u \in U)$ .

Il s'agit donc du programme linéaire suivant :

$$(MM) \begin{cases} \text{Max } Z = z^1 + z^2 + \dots + z^K \\ \text{Sous les contraintes :} \\ A \cdot \varphi^k - z^k \cdot b^k = 0 & (k = 1, \dots, K) \\ \sum_{k=1}^K \varphi_u^k \leq c_u & (\forall u \in U) \\ \varphi_u^k \geq 0 & (\forall k, \forall u). \end{cases}$$

#### Problème du multiflot compatible (MC) sur un graphe muni de bornes supérieures de capacité

Contrairement au problème précédent, la valeur de chacun des flots est maintenant une donnée imposée  $d^k$  (demande entre  $s_k$  et  $t_k$ ).

Le problème est de trouver un multiflot compatible avec les capacités  $c_u$  sur les arcs ; autrement dit de résoudre l'équations et d'inéquations linéaires :

$$(MC) \begin{cases} A \cdot \varphi^k = d^k \cdot b^k & (k = 1, \dots, K) \\ \sum_{k=1}^K \varphi_u^k \leq c_u & (\forall u \in U) \\ \varphi_u^k \geq 0 & (\forall k, \forall u). \end{cases}$$

#### Problème du multiflot compatible et de coût minimum (MCM)

Comme ci-dessus, la valeur de chacun des flots est imposée (demande à satisfaire) et chaque arc  $u \in U$  est muni d'une capacité  $c_u \geq 0$ .

On suppose, de plus, que l'on se donne sur chaque arc  $u$ , le coût de passage d'une unité de flot  $\gamma_u$ .

(Dans les applications pratiques on a généralement  $\gamma_u \geq 0 (\forall u)$  ; cependant nous nous contenterons ici de faire l'hypothèse, plus générale, que  $\sum_{u \in \mu} \gamma_u \geq 0$  pour tout circuit  $\mu$ .)

On recherche alors, parmi l'ensemble des multiflots compatible, un multiflot  $\Psi$  de coût  $\sum_{u \in \mu} \gamma_u \cdot \Psi_u$  minimal.

Ceci revient à résoudre le programme linéaire :

$$(MCM) \left\{ \begin{array}{l} \text{Min } \gamma \cdot \Psi = \sum_{u \in U} \gamma_u \cdot \Psi_u \\ \text{Sous les contraintes :} \\ A \cdot \varphi^k = d^k \cdot b^k \quad (k = 1, \dots, K) \\ \Psi_u = \sum_{k=1}^K \varphi_u^k \leq c_u \quad (\forall u \in U) \\ \varphi_u^k \geq 0 \quad (\forall k, \forall u). \end{array} \right.$$

Les formulations précédentes des problèmes de multiflots font intervenir la matrice d'incidence sommets-arcs du graphe. Nous dirons qu'il s'agit d'une formulation sommets-arcs des problèmes de multiflots.

**remarque 1.2.** *La dimension d'un problème de multiflots en formulation sommets-arcs est donc, a priori,  $K$  fois celle d'un flot simple ( $KM$  variables,  $KN + M$  contraintes).*

*Cependant, si  $K > N$ , il est important de remarquer que l'on peut toujours se ramener à  $N$  flots simples en considérant l'ensemble des flots ayant une même origine comme un seul flot simple, d'où au plus  $NM$  variables et  $N^2 + M$  contraintes.*

### 1.3.2 Formulation arcs-chemins des problèmes de multiflot

Nous allons montrer maintenant qu'on peut donner une autre formulation de ces mêmes problèmes, utilisant des matrices d'incidence arcs-chemins, on parlera alors de formulation arcs-chemins.

Commençons par une remarque. Soit  $\Psi = (\Psi_u)$  une solution optimale de(MM) ou de(MCM) ou, dans le cas de (MC), simplement une solution ; et supposons que le support d'un des flots  $\varphi^k$  (l'ensemble des arcs  $u$  tels que  $\varphi_u^k > 0$ ) contienne un circuit  $\mu$  (dans le cas du MCM, ce circuit est nécessairement de coût nul).

Soit  $\vec{\mu}$  le vecteur caractéristique associé à ce circuit ( $\vec{\mu} = +1$  si est seulement si  $u \in \mu$ ) et considérons le flot

$$\varphi^{,k} = \varphi^k - \varepsilon \vec{\mu}.$$

Choisissons  $\varepsilon > 0$  de façon à ce qu'une composante de  $\varphi^k$  au moins s'annule (les autres restant  $\geq 0$ ).

En remplaçant le flot  $\varphi^k$  par le flot  $\varphi^{\prime k}$  :

- on ne change pas la valeur  $z^k$  du flot ;
- on obtient un multiflots qui est encore une solution de (MC) ou solution optimale de (MM) ou (MCM).

On en déduit que l'on peut résoudre chacun de ces problèmes en imposant aux flots  $\varphi^k$  d'avoir des supports sans circuit. d'après le théorème de décomposition de flot, que chacun des flots  $\varphi^k$  peut se décomposer et une somme de flots élémentaires s'écoulant sur des chemins élémentaires entre  $s_k$  et  $t_k$ . Cette remarque est à la base des formulations arcs-chemins des problèmes de multiflots. Appelons  $p(k)$  le nombre de chemins élémentaires distincts entre  $s_k$  et  $t_k$  dans  $G$ , et pour  $j = 1, \dots, p(k)$ , notons  $Q_j^k$  le  $j$ -ième chemin de la liste. définissons alors le vecteur caractéristique de  $Q_j^k$ , c'est-à-dire :

$$\begin{aligned} (p_j^k)_u &= +1 \text{ si } u \in Q_j^k \\ (p_j^k)_u &= 0 \text{ sinon.} \end{aligned}$$

Enfin désignons par  $x_j^k \geq 0$  (nombre réel) la quantité de flot  $k$  circulant sur la chemin  $Q_j^k$ .

on en déduit les nouvelles formulations suivantes :

### Multiflot maximum

$$(MM') \left\{ \begin{array}{l} \text{Max } \sum_{k=1}^K \sum_{j=1}^{p(k)} x_j^k \\ \text{Sous les contraintes :} \\ \sum_{k=1}^K \sum_{j=1}^{p(k)} p_j^k \cdot x_j^k \leq c \\ x_j^k \geq 0 \end{array} \right. \quad (\forall k = 1, \dots, K; \forall j = 1, \dots, p(k))$$

où  $c = (c_1, c_2, \dots, c_M)^T$  désigne le vecteur des capacités (le signe  $\leq$  entre deux vecteurs désigne l'inégalité composante par composante).

### Multiflot compatible

$$(MC') \left\{ \begin{array}{l} \text{Trouver } (k=1, \dots, K; j=1, \dots, p(k)) \\ \text{Vérifiant :} \\ \sum_{k=1}^K \sum_{j=1}^{p(k)} p_j^k \cdot x_j^k \leq c \\ \sum_{j=1}^{p(k)} x_j^k = d^k \end{array} \right. \quad (k = 1, \dots, K)$$

### Multiflot compatible et de coût minimum

En posant  $\gamma_j^k = \sum_{u \in Q_j^k} \gamma_u$  (coût du chemin  $Q_j^k$ ) le problème est :

$$(MCM') \left\{ \begin{array}{l} \text{Max } \sum_{k=1}^K \sum_{j=1}^{p(k)} \gamma_j^k \cdot x_j^k \\ \text{Sous les contraintes :} \\ \sum_{k=1}^K \sum_{j=1}^{p(k)} p_j^k \cdot x_j^k \leq c \\ \sum_{j=1}^{p(k)} x_j^k = d^k \quad (k = 1, \dots, K) \\ x_j^k \geq 0 \quad (\forall k = 1, \dots, K; \forall j = 1, \dots, p(k)) \end{array} \right.$$

En associant à chaque arc  $u$  une variable d'écart  $s_u \geq 0$  (différence entre la capacité  $c_u$  et la valeur du multiflot  $\Psi_u$ ) le problème est parfaitement équivalent à la maximalisation de la valeur des capacités :

$$(MCM'') \left\{ \begin{array}{l} \text{Max } \sum_{u \in U} \gamma_u \cdot x_u \\ \text{Sous les contraintes :} \\ \sum_{k=1}^K \sum_{j=1}^{p(k)} p_j^k \cdot x_j^k + s = c \\ \sum_{j=1}^{p(k)} x_j^k = d^k \quad (k = 1, \dots, K) \\ x_j^k \geq 0, s_u \geq 0 \end{array} \right.$$

où  $s = (s_1, s_2, \dots, s_M)$  est le vecteur des variables d'écart.

On voit que les différentes formulations des problèmes de multiflots

(sommets-arcs ou arcs-chemins) que nous avons données font intervenir la programmation linéaire.

### 1.3.3 Étude du problème du multiflot compatible

Nous commencerons par établir une condition nécessaire et suffisante pour l'existence d'un multiflot compatible. Puis nous donnerons un algorithme permettant de vérifier cette condition.

La formulation arcs-chemins du problème du multiflot compatible peut aussi s'écrire :

$$(MC'') \left\{ \begin{array}{l} \text{Trouver des variables } x_j^k \geq 0 \\ \text{vérifiant :} \\ \sum_{k=1}^K \sum_{j=1}^{p(k)} p_j^k \cdot x_j^k \leq c \\ \sum_{j=1}^{p(k)} x_j^k \geq d^k \\ x_j^k \geq 0 \end{array} \right.$$

D'une façon équivalente, il s'agit de trouver une solution au système d'inéquations linéaires :

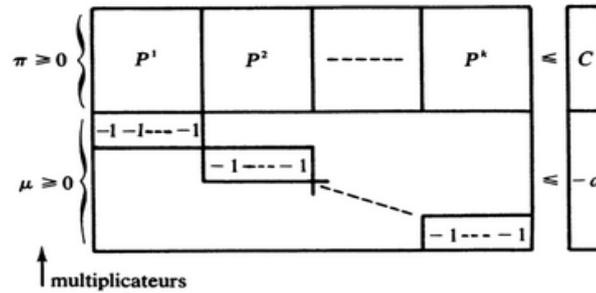
$$\left\{ \begin{array}{l} \sum_{k=1}^K \sum_{j=1}^{p(k)} p_j^k \cdot x_j^k \leq c \quad (1.1) \\ - \sum_{j=1}^{p(k)} x_j^k \leq -d^k \quad (1.2) \\ x_j^k \geq 0 \end{array} \right.$$

Rappelons le lemme de Farkas et Minkowski :

Soit A une matrice réelle ( $m * n$ ) et b m-vecteur. Le système d'inéquations linéaires :  $Ax \leq b$  a une solution non négative ( $x \geq 0$ ) si et seulement si :

$$\forall y \in \mathbb{R}^m (y \geq 0) : y^T \cdot A \geq 0 \Rightarrow y^T \cdot b \geq 0.$$

Associons à chacune des contraintes(1.1) un multiplicateur  $\pi_u \geq 0$ , et à chacune des contraintes :(1.2) un multiplicateur  $\mu^k \geq 0$ .



$\forall k = 1, \dots, K$ , notons  $P^k$  la matrice  $[P_1^k, P_2^k, \dots, P_{p(k)}^k]$ .  
 Il faut et il suffit de vérifier que :

$$\text{Condition}(C) \begin{cases} \forall \pi \geq 0, \forall \mu \geq 0 \\ \pi \cdot P^k - \mu^k \geq 0 (\forall k) \Rightarrow \pi - \mu \cdot d \geq 0. \end{cases}$$

On observe que :

$$\pi \cdot P^k - \mu^k \geq 0 (\forall k) \Rightarrow \mu^k \leq \min_{1 \leq j \leq p(k)} \{ \pi \cdot P_j^k \}.$$

Pour que la condition (C) soit vérifiée, il faut et il suffit qu'elle le soit dans le cas le plus défavorable, c'est-à-dire (puisque  $d \geq 0$ ) lorsque :  
 $\mu^k = \min_j \{ \pi \cdot P_j^k \}$  = longueur du plus court chemin (au sens des longueur  $\pi_u \geq 0$ ) entre  $s_k$  et  $t_k$ .

On en déduit :

**Théorème 1.1.** Une condition nécessaire et suffisante pour qu'il existe un multiflot compatible est que :

$$\forall \pi = (\pi_1, \pi_2, \dots, \pi_M) :$$

$$\mu \cdot d \leq \pi \cdot c \tag{1.3}$$

où :

$$\mu = (\mu^1, \mu^2, \dots, \mu^K)$$

avec :

$\mu^k =$  longueur du plus court chemin (au sens des longueurs  $\pi_u \geq 0$ ) entre  $s_k$  et  $t_k$ .

Il est intéressant de remarquer que cette condition est absolument analogue au théorème du flot maximum et de la coupe minimale pour les flots simples. Pour le voir il suffit de considérer un seul flot ( $K = 1$ ) et de prendre comme vecteur  $\pi$  l'ensemble des vecteurs caractéristiques des coupes  $\omega^+(A)$  séparant  $s_1$  et  $t_1$  dans le graphe. On obtient la condition : pour toute coupe séparant  $s_1$  et  $t_1$  :

$$\text{valeur du flot} \leq \text{capacité de la coupe.}$$

Ce rapprochement illustre également la difficulté des problèmes de multiflots : alors que dans le cas de flots simples, il suffit de vérifier la condition (1.3) pour un nombre fini de vecteurs  $\pi$  à composantes 0 ou 1, dans le cas de multiflot la même condition doit être vérifiée pour un nombre a priori infini de vecteurs  $\pi$  à composantes réelles quelconques. (En réalité, on pourra remarquer qu'en résolvant le problème par programmation linéaire, le nombre de vecteurs  $\pi$  à examiner n'est pas infini puisqu'on sait qu'on peut se restreindre aux points extrêmes d'un polyèdre convexe, lesquels sont en nombre fini et à coordonnées rationnelles.

Nous allons voir maintenant comment on peut déduire du théorème précédent un algorithme pour vérifier l'existence d'un multiflot compatible.

Remarquons, tout d'abord, que si l'on multiplie tous les nombres  $\pi_u$  par une même constante  $\lambda$ , tous les  $\mu^k$  sont multipliés par cette constante  $\lambda$ .

Il est donc légitime d'imposer une condition de normalisation sur les  $\pi_u$  qui peut être, par exemple :

$$\pi \cdot v = \sum_u \pi_u = 1.$$

(Par convention,  $v$  est le  $M$ -vecteur dont toutes les composantes sont égales à 1.)

La méthode consiste alors à rechercher le maximum  $L^*$  de la fonction

$$\text{Problème (P)} \begin{cases} L(\pi) = \mu - \pi c \\ \text{sous les contraintes :} \\ \pi \cdot v = 1, \pi \geq 0. \end{cases}$$

D'après le théorème précédent :

- si  $L^* \leq 0$  alors il existe un multiflot compatible ;
- si  $L^* > 0$  alors le problème n'a pas de solution.

Pour résoudre le problème (P) il nous faut introduire la notion de routage et de vecteur associé à un routage.

**Définition 1.19.** *On dit que l'on a effectué un routage des flots  $\varphi^1, \varphi^2, \dots, \varphi^K$  lorsqu'on a affecté à chaque flot  $k$  un chemin particulier  $Q_{\sigma(k)}^k$  pris parmi l'ensemble des flots dans  $p(k)$  chemins entre  $s_k$  et  $t_k$  ( $1 \leq \sigma(k) \leq p(k)$ ).*

*Autrement dit, un routage est une application  $\sigma$  de l'ensemble des flots dans l'ensemble des chemins.*

*Étant donné un routage  $\sigma$ , on appelle vecteur associé à  $\sigma$  le vecteur :*

$$R^\sigma = \sum_{k=1}^K P_{\sigma(k)}^k \cdot d^k.$$

C'est donc le vecteur de composants  $(R_u^\sigma)_{u \in U}$ . où  $R_u^\sigma$  est la somme totale des flots sur l'arc  $u$  lorsque chaque flot  $\varphi^k$  s'écoule en totalité sur le chemin  $Q_{\sigma(k)}^k$ .

Il est important de remarquer que le nombre de routages distincts est  $\prod_{i=1}^K p(i)$ , et, par conséquent, l'ensemble des tous les routages  $\mathcal{R}$  est un ensemble fini.

Considérons, pour  $\pi \geq 0$  donné, ( $\pi \cdot v = 1$ ) le routage particulier  $\sigma_\pi$  défini par :

$$\pi \cdot P_{\sigma_\pi(k)}^k = \min_{j=1, \dots, p(k)} \{ \pi \cdot P_j^k \} = \min_{\sigma} \{ \pi \cdot P_{\sigma(k)}^k \}.$$

Remarquer que, pour  $\pi$  donné, le calcul du vecteur  $R^\sigma$  nécessite seulement la détermination, pour  $k = 1, \dots, K$ , des plus courts chemins

de  $s_k$  et  $t^k$  relativement aux longueurs  $\pi_u$  sur les arcs. Il existe pour cela de nombreux algorithmes, par exemple l'algorithme de Floyd, ou celui de Dantzig.

La valeur de la fonction L au point  $\pi$  est alors :

$$L(\pi) = \sum_{k=1}^K \mu^k \cdot d^k - \pi \cdot c = \sum_{k=1}^K \mu^k \cdot P_{\sigma_\pi(k)}^k - \pi \cdot c$$

d'où :

$$L(\pi) = \pi \cdot (R^{\sigma_\pi} - c) = \min_{\sigma \in \mathcal{R}} \{ \pi \cdot (R^\sigma - c) \}.$$

Cette dernière expression montre que  $L(\pi)$  est une fonction définie sur le simplexe

$$S = \{ \pi / \pi \geq 0, \pi \cdot v = 1 \}.$$

comme l'enveloppe inférieure d'un nombre fini (en pratique très grand) d'hyperplans d'équations :

$$t = \pi \cdot (R^\sigma - c) \quad (\forall \sigma \in \mathcal{R}).$$

Il est facile de démontrer que  $L(\pi)$  est une fonction concave non partout différentiable, et que le vecteur  $R^\sigma - c$  est un sous-gradient de L et  $\pi$  (on rappelle que, pour une fonction  $L(\pi)$  concave non partout différentiable,  $D \in \mathbb{R}^M$  est un sous-gradient en  $\pi$  si et seulement si :

$$(\forall \pi' \in \mathbb{R}^M : L(\pi')(\pi) + (\pi' - \pi) \cdot D).$$

On peut alors résoudre le problème (p) par un algorithme de sous-gradient.

**algorithme.**

a) Choisir une solution de départ  $\pi^0$ . (Par exemple  $\frac{1}{M}(1, 1, \dots, 1)$ ).

$$t \leftarrow 0;$$

b) **Répéter**

À l'étape courante t,  $\pi^t$  est la solution courante. Déterminer le vecteur  $R^{\sigma^*}$  défini par :

$$\pi^t \cdot R^{\sigma^*} = \min_{\sigma \in \mathcal{R}} \{\pi^t \cdot R^\sigma\}$$

(Ceci requiert la détermination d'un ensemble de plus courts chemins.)

**Si** ( $L(\pi^t) > 0$ ) **alors**

interrompre le calcul : on a obtenu la preuve qu'il n'existe pas de solution admissible.

**Sinon**

Calculer  $D^t = R^{\sigma^*} - c$ , un sous-gradient de  $L$  en  $\pi^t$ , puis  $\pi^{t+1}$ , projection sur  $S$  du vecteur  $\pi^t + \lambda_t D^t$  (où  $\lambda_t$  est le pas de déplacement choisi pour l'itération  $t$ ).

**Fsi**

$$t \leftarrow t + 1;$$

**Tant que** (condition d'arrêt non vérifiée).

La procédure converge vers l'optimum  $L^* = L(\pi^*)$  à condition de choisir les pas de déplacement  $\lambda_t$  vérifiant :

$$\left\{ \begin{array}{l} \lambda_t \rightarrow 0 \quad (t \rightarrow +\infty) \\ \sum_0^\infty \lambda_t = +\infty \end{array} \right.$$

Si on choisit :

$$\lambda_t = \rho_t \frac{-L(\pi^t)}{\|D^t\|^2}$$

avec  $0 < \rho_t \leq 2$  (coefficient de relaxation) on peut établir que :

- S'il n'existe pas de multiflot compatible, c'est-à-dire  $L(\pi^*) > 0$ , la procédure permet d'obtenir en un nombre fini d'itérations un point  $\bar{\pi}$  tel que  $L(\bar{\pi}) \geq 0$ .
- S'il existe un multiflot compatible, alors la procédure converge vers  $\pi^*$  (mais pas nécessairement de façon finie) à condition de choisir :

$$\rho_t \rightarrow 0^+ \quad (t \rightarrow +\infty).$$

On voit que la méthode ne permet de conclure avec certitude, en un nombre fini d'itérations, que dans le cas où il n'existe pas de multiflot compatible. Cependant, lorsque, à une étape quelconque, la plus grande valeur de  $L(\pi^*)$  en résolvant de façon exacte le programme linéaire :

$$(PL) \left\{ \begin{array}{l} \bar{L} = \max_{L, \pi} L \\ \text{Sous les contraintes} \\ \pi \cdot (R^\sigma - c) - L \geq 0 \quad (\text{pour } \sigma \in \bar{\mathcal{R}}) \\ \pi \cdot v = 1, \pi \geq 0 \end{array} \right.$$

(en pratique, on résoudra plutôt le dual de (PL) qui comporte généralement beaucoup moins de contraintes que de variables).

Ici  $\bar{\mathcal{R}}$  désigne le sous-ensemble de routage obtenus successivement jusqu'à l'étape considérée de l'algorithme. On obtient ainsi un encadrement de l'optimum souvent de bonne qualité :

$$L(\pi^t) \leq L(\pi^*) \leq \bar{L}$$

et, dans le cas où  $L\bar{L} \leq 0$ , on peut conclure avec certitude à l'existence d'un multiflot compatible.

Le seul cas où on ne peut conclure est celui où :

$$L(\pi^t) \leq 0 \leq \bar{L}.$$

Il faut alors poursuivre les itérations (à moins que l'écart  $\bar{L} - L(\pi^t)$  ne soit suffisamment faible pour admettre que  $\pi^t$  est une bonne approximation de la solution optimale, compte tenu de la précision recherchée).

La résolution du programme linéaire (PL) peut souvent être évitée en pratique, en pratique lorsqu'on ne désire qu'une réponse (tout ou rien) et lorsqu'on sait que les données du problème ne sont pas connues avec un degré de finesse extrême : c'est le cas dans de nombreuses applications pratiques.

Cependant, lorsqu'on s'intéresse à la détermination effective d'un multiflot compatible et de ses composantes  $\varphi^1, \varphi^2, \dots, \varphi^K$ , nous allons voir que la résolution de (PL) (en fait, de son dual) est très intéressante.

Écrivons le programme dual de (PL) en associant à chaque contrainte de type  $(R^\sigma - c) - L \geq 0$  une variable duale  $y_\sigma \geq 0$ , et à la contrainte  $\pi \cdot v = 1$  la variable duale  $\eta$ . On obtient :

$$\left\{ \begin{array}{l} \bar{\eta} = \text{minimum de } \eta \\ \text{Sous les contraintes} \\ \sum_{\sigma \in \bar{\mathcal{R}}} R^\sigma \cdot y_\sigma - \eta v \\ \sum_{\sigma \in \bar{\mathcal{R}}} y_\sigma = 1 \\ y_\sigma \geq 0 \quad (\forall \sigma \in \bar{\mathcal{R}}). \end{array} \right.$$

On sait que a à l'optimum :  $\bar{L} = \bar{\eta}$

et lorsque  $\bar{\eta}$  est négatif ou nul, on a :

$$\sum_{\sigma \in \bar{\mathcal{R}}} R^\sigma \cdot y_\sigma \leq c \quad \text{avec : } \sum_{\sigma \in \bar{\mathcal{R}}} y_\sigma = 1.$$

Les valeurs  $y_\sigma$  permettent alors de construire le multiflot compatible suivant :

Chacun des flots  $\varphi^k (k = 1, \dots, K)$  est obtenu en considérant successivement tous les routages  $\sigma \in \bar{\mathcal{R}}$  et, pour chaque  $\sigma$ , en faisant circuler sur le chemin  $\mathcal{R}_{\sigma(k)}^k$  une quantité de flot égale à  $y_\sigma \cdot d^k$ .

Remarquons, pour terminer, que le problème du multiflot compatible peut également être considéré comme un problème de multiflot de coût minimum particulier. L'algorithme du paragraphe ci-dessous constitue donc une seconde méthode de résolution des problèmes de multiflots compatibles.

### 1.3.4 Une méthode de décomposition par affectation de capacités

Il existe de nombreuses méthodes pour la résolution des problèmes de multiflots de coût minimum. Dans leur quasi-totalité elles utilisent la programmation linéaire ou constituent des adaptations de l'algorithme du simplexe. On peut citer :

- La génération de colonnes (Ford et Fulkerson, 1958) et la décomposition par les prix de Dantzig-Wolfe (Hu, 1963 ; Jarvis, 1969) ;
- L'algorithme primal-dual (Jewell, 1958 ; 1966) ;
- Les techniques de partitionnement (Grigoriadis et White, 1972) ou la représentation compacte de l'inverse de la base (Generalized upper bounding, Saigal, 1968) ;
- La décomposition par les ressources (Right hand-side allocation, Sakarovitch, 1966 ; Lasdon, 1970 ; Geoffrion, 1972 ; Kennington et Shalaby ; 1977).

Pour un panorama très complet de toutes ces méthodes, on pourra se reporter à Assad (1978) et à Kennington (1978).

L'algorithme que nous allons présenter se rattache à la dernière famille (décomposition par les ressources), mais, contrairement aux autres méthodes de ce type, elle utilise un algorithme de sous-gradient, au lieu de la programmation linéaire, pour résoudre le programme principal. le problème à résoudre en formulation sommets-arcs est le suivant :

$$(MCM) \left\{ \begin{array}{l} \text{Min } \sum_{u \in U} \sum_{k=1}^K \gamma_u \cdot \varphi_u^k \\ \text{Sous les contraintes :} \\ A \cdot \varphi^k = d^k \cdot b^k \quad (k = 1, \dots, K) \\ \sum_{k=1}^K \varphi_u^k \leq c_u \quad (u \in U) \\ \varphi_u^k \geq 0. \end{array} \right.$$

En associant à chaque contrainte de demande  $k$  une variable artificielle  $\alpha^k \geq 0$  ( $k=1, \dots, K$ ) le problème se met sous la forme équivalente :

$$\left\{ \begin{array}{l} \text{Min } \sum_{u \in U} \sum_{k=1}^K \gamma_u \cdot \varphi_u^k + H \cdot \sum_{k=1}^K \alpha^k \\ \text{Sous les contraintes :} \\ A \cdot \varphi^k + \alpha^k \cdot b^k = d^k \cdot b^k \quad (k = 1, \dots, K) \\ \sum_{k=1}^K \varphi_u^k \leq c_u \quad (u \in U) \\ \varphi_u^k \geq 0, \alpha^k \geq 0. \end{array} \right.$$

où  $H$  (coût des variables artificielles) est une constante positive suffisamment grande pour qu'à l'optimum, toutes les variables artificielles soient nulles (lorsqu'il existe un multiflot compatible).

Nous allons maintenant décomposer le problème en affectant à chaque flot  $\varphi^k$  une fraction  $y_u^k$  de la capacité totale disponible sur l'arc  $u$ , en imposant évidemment :

$$\sum_{k=1}^K y_u^k = c_u \quad (\forall u \in U)$$

On note  $y^k = [y_u^k]_{u \in U}$ .

Pour une affectation  $y = (y^1, y^2, \dots, y^K)$  quelconque, définissons la fonction

$g(y)$  par :

$$g(y) = g(y^1, y^2, \dots, y^K) = \sum_{k=1}^K g^k(y^k)$$

où les fonctions  $g^k(y^k)$  sont les valeurs optimales des problèmes de flot simple à coût minimum(programmes satellites) :

$$(PS^K) \left\{ \begin{array}{l} g^k(y^k) = \text{minimiser} \sum_{k=1}^K \gamma_u \cdot \varphi_u^k + H \cdot \alpha^k \\ \text{Sous les contraintes :} \\ A \cdot \varphi^k + \alpha^k \cdot b^k = d^k \cdot b^k \\ 0 \leq \varphi^k \leq y^k. \end{array} \right.$$

On voit que, pour une affectation de capacités  $y = (y^1, y^2, \dots, y^K)$  donnée, la fonction  $g(y)$  se calcule très simplement K problèmes indépendante, c'est-à-dire d'un vecteur  $y = (y^1, y^2, \dots, y^K)$  tel que :

$$(PP) \left\{ \begin{array}{l} g(y) = \text{minimiser} \\ \text{Sous les contraintes :} \\ \sum_{k=1}^K y^k = c, y^k \geq 0 \end{array} \right.$$

(programmation principal).

On démonte que :

1.  $g(y)$  est une fonction convexe, non partout différentiable, des variables  $y_u^k$  ;
2. pour  $k = 1, \dots, K$  notons  $\theta^k = (\theta_u^k)_{u \in U}$  les variables duales associées aux contraintes  $\varphi^k \leq y^k$  à l'optimum de  $(PS^K)$ . Alors le vecteur  $\theta = (\theta^k)_{k=1, \dots, K}$  est un sous-gradient de  $g$  en  $y$  c'est-à-dire :

$$\forall y' \geq 0 : g(y') - g(y) \geq \theta^T \cdot (y' - y)$$

Le problème peut donc être résolu par toute méthode d'optimisation de fonction convexe non différentiable, par exemple par un algorithme de sous-gradient analogue à celui utilisé pour le problème du multiflot compatible.

Cependant, comme cet algorithme ne permet d'obtenir, en un nombre fini d'itérations, qu'une solution approchée du problème, il est nécessaire de pouvoir évaluer, à tout instant, la distance entre la solution courante et la

solution optimale, de façon à ne pas arrêter les itérations trop tôt. Il suffit, pour cela, de résoudre le problème dual de (MCM) de façon approchée pour en déduire une borne inférieure  $g$ . Cette borne inférieure permettra alors d'interrompre les calculs dès que la différence avec la solution courante devient inférieure en valeur relative à une certaine tolérance  $\varepsilon$  fixée à l'avance. Ou bien, dans le cas où les calculs doivent être interrompus (par exemple lorsque la limite de temps calcul est atteinte), elle permet d'évaluer la distance à l'optimum et de savoir si la solution obtenue peut être considérée comme acceptable.

La méthode est alors la suivante :

**algorithme.**

a) Déterminer un minorant  $\underline{g}$  par résolution approchée du problème dual (en utilisant une méthode de sous-gradient).

$t \leftarrow 0$  :

Pour  $k = 1, \dots, K$ , résoudre les problèmes  $(PS^k)$  avec  $y^k = c$ .

**Si**, dans la solution optimale d'un des problèmes  $(PS^k)$ , une des variables artificielles  $\alpha^k$  est non nulle, **alors** on a obtenu la preuve qu'il n'existe pas de multiflot compatible et la procédure se termine. **Si non**, soient  $\varphi^k$  ( $k = 1, \dots, K$ ) les flots optimaux obtenus.

Déterminer une première affectation de capacités  $\bar{y}$  en projetant le vecteur  $\bar{\varphi} = (\bar{\varphi}^1, \bar{\varphi}^2, \dots, \bar{\varphi}^K)$  sur le sous-ensemble (convexe)

$$S = \{y / y \geq 0, \sum_{k=1}^K y^k = c\}$$

b) **Répéter**

À l'étape courante  $t$ ,  $\bar{y}$  est l'affectation de capacité courante.

Déterminer

$$g(\bar{y}) = \sum_{k=1}^K g^k(\bar{y}^k)$$

en résolvant successivement les  $K$  problèmes satellites  $(PS^k)$  ( $k = 1, \dots, K$ ).

Soient  $\bar{\theta}^k$  les variables duales optimales obtenues.

**Si**  $(g(\bar{y}) - \underline{g}) \leq \varepsilon \underline{g}$  **alors**

La procédure se termine, la solution étant

epsilon-optimale ( $\varepsilon$  est un paramètre  $> 0$  fixé à l'avance).

**Sinon**

Déterminer une nouvelle affectation de capacités en se déplaçant dans la direction opposée au sous-gradient  $\bar{\theta} = (\bar{\theta}^1, \bar{\theta}^2, \dots, \bar{\theta}^K)$ . Pour cela, calculer  $\bar{y}$  comme la projection du vecteur  $\bar{y} - \lambda_t \bar{\theta}$  sur

$$S = \{y/y \geq 0, \sum_{k=1}^K y^k = c\}$$

puis faire :

$$\bar{y} \leftarrow \bar{y};$$

$$t \leftarrow t + 1;$$

**FSi**

**Tant que** (condition d'arrêt non vérifiée).

Pour le choix des pas de déplacement, on utilisera les règles déjà évoquées dans §1.3.3 . la condition d'arrêt pourra être liée à la longueur des pas de déplacement  $\lambda_t$  ou, plus simplement, définie par une limite sur le nombre d'itérations.

Notons que si le minorant  $\underline{g}$  obtenu en a) est de qualité suffisamment bonne (par rapport à la tolérance  $\epsilon$ ), la terminaison de la procédure se fera lorsque la condition  $g(\bar{y}) - \underline{g} \leq \epsilon \underline{g}$  est obtenue.

Pour la résolution approchée du problème dual de (MCM)qui permet d'obtenir en a) un minorant  $\underline{g}$ , on pourra procéder par décomposition, en utilisant de nouveau une méthode de sous-gradient.

On associera à chaque contrainte de capacité

$$\sum_{k=1}^K \varphi_u^k \leq c.$$

une variable de Lagrange  $\pi_u \geq 0$ , et on définira la fonction duale  $L(\pi)$  par :

$$\left\{ \begin{array}{l} L(\pi) = \min_{\varphi} \sum_{k=1}^K \sum_{u \in U} (\gamma_u + \pi_u) \cdot \varphi_u^k - \pi \cdot c \\ \text{Sous les contraintes :} \\ A \cdot \varphi^k = d^k \cdot b^k, \quad (k = 1, \dots, K) \\ \varphi_u^k \geq 0. \end{array} \right.$$

Le calcul de  $L(\pi)$  pour  $\pi$  donné se décompose donc en K problèmes de flots simples à coût minimum sans contrainte :

$$\left\{ \begin{array}{l} \min_{\varphi^k} \sum_{u \in U} (\gamma_u + \pi_u) \cdot \varphi_u^k \\ \text{Sous les contraintes :} \\ A \cdot \varphi^k = d^k \cdot b^k \\ \varphi_u^k \geq 0. \end{array} \right.$$

Dans la solution est obtenue en recherchant un plus court chemin entre  $s_k$  et  $t_k$  dans  $G$  (chaque arc  $u \in U$  étant muni d'une longueur  $\gamma_u + \pi_u$ ) et en faisant passer sur ce chemin  $d^k$  unités de flot.

Soit  $\sigma_\pi$  le routage obtenu pour le flot  $k$ . Désignons par  $R^{\sigma_\pi}$  le vecteur associé §1.3.3. On sait que  $L(\pi)$  est un minorant de l'optimum de (MCM), le problème dual (D) consiste à rechercher le meilleur des minorants de ce type :

$$\left\{ \begin{array}{l} \text{Max } L(\pi) \\ \pi \in \mathbb{R}_+^M. \end{array} \right.$$

D'autre part, comme  $(R^{\sigma_\pi} - c)$  est un sous-gradient de  $l$  au point  $\pi$ , l'utilisation d'un algorithme de sous-gradient permet d'obtenir facilement une borne inférieure  $\underline{g} = L(\bar{\pi})$  approchant d'aussi près que l'on veut l'optimum  $g^*$  de (MCM).

**remarque 1.3.** À l'étape  $a$ ) et à l'étape courante de l'algorithme on doit rechercher la projection d'un point  $\xi$  de coordonnées  $\xi = (\xi_u^k)$  sur le polyèdre

$$S = \{y \geq 0 / \sum_{k=1}^K y^k = c_u\}.$$

Ceci revient à résoudre le problème :

$$\left\{ \begin{array}{l} \text{Miniser } \sum_{u \in U} \sum_{k=1}^K (y_u^k - \xi_u^k)^2 \\ \text{Sous les contraintes :} \\ \sum_{k=1}^K y_u^k = c_u, \quad (\forall u \in U) \\ y_u^k \geq 0, \quad (\forall u, \forall k). \end{array} \right.$$

lequel se décompose en  $M$  sous-problèmes indépendants (un pour chaque arc  $u \in U$ ) du type :

$$\left\{ \begin{array}{l} \text{Min} \sum_{i=1}^n (x_i - \bar{x}_i)^2 \\ \text{Sous les contraintes :} \\ \sum_{i=1}^n x_i = x_0 \quad (x_0, \text{donné}) \\ y_i^k \geq 0, \quad \forall i = 1, \dots, n). \end{array} \right.$$

On en détermine facilement la solution optimale, de façon explicite, en utilisant les conditions de Kuhn et Tucker.

# Chapitre 2

## Les Réseaux

### Introduction

Un réseau est un graphe orienté  $G = (N, A)$  avec une valuation positive de ses arcs. La valuation  $c(x, y)$  est appelée la capacité de l'arc. On distingue sur  $N$  deux sommets particuliers, une source  $s$  et une destination  $t$ , les autres sommets sont les noeuds intermédiaires du réseaux.

Dans ce chapitre, nous allons étudier le problème de conception de réseaux, qui concerne la définition de la topologie d'un réseau permettant d'interconnecter un ensemble de sites/noeuds aux caractéristiques différentes.

En général, cette topologie de réseau doit satisfaire à certains standards de fiabilité et de performance, et ce à moindre coûts. Le processus de conception comprend également les procédures de routage ainsi que le dimensionnement des lignes et noeuds.

Dans ce chapitre nous présentons toute la procédure théorique de construction du modèle mathématique. Enfin nous exhibons le programme linéaire en variables bivalentes que l'on exploite en vue de résoudre le problème.

### 2.1 Le réseau de Transport

Un réseau de transport [1] est un graphe sans boucles, ou chaque arc est value par un nombre positif  $c(u) \geq 0$ , appelé "capacite" de l'arc  $u$ . En outre, un tel réseau verifie les hypotheses suivantes :

- Il existe un seul noeud  $s$  qui n'a pas de prédécesseurs, tous les autres en ont au moins un. Ce noeud est appelé l'entrée du réseau, ou la source.
- Il existe également un seul noeud  $t$  qui n'a pas de successeurs, tous les autres en ont au moins un. Ce noeud est appelé la sortie du réseau, ou le puits.

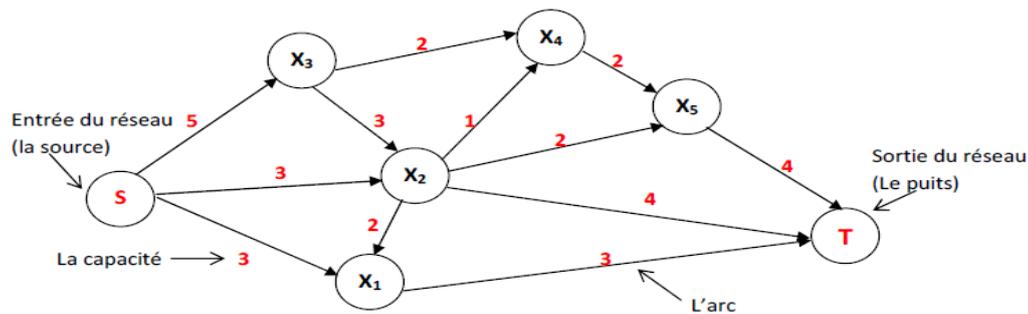


FIGURE 2.1 – Un réseau et ses capacités

**remarque 2.1.** *Le problème de transport consiste à transporter une certaine marchandise depuis des points (sources) d'un réseau à d'autres points (puits) d'un réseau. Le transport de marchandises à travers le réseau a un coût connu, nous cherchons donc à satisfaire la demande en minimisant le coût total du transport.*

*Dans le problème de transport, le réseau peut être représenté par un graphe bi-parti à deux étages : l'étage des sources et l'étage des puits. Il n'y a pas de point intermédiaire dans le réseau, c'est à dire chaque point est soit une source soit un puits. De plus, chaque source est reliée à chaque puits et il n'y a pas de chemin d'une source à une autre ni d'un puits à un autre.*

## 2.2 Le réseau de télécommunication

Les réseaux de télécommunication sont naturellement représentés sous forme de graphes. Ils constituent ainsi un terrain d'application très fertile à la théorie des graphes. Les problèmes de multiflot se sont ainsi imposés comme l'un des éléments essentiels des problèmes d'optimisation de réseaux de télécommunication [14].

**Définition 2.1.** *Un réseau de télécommunication est un réseau d'arcs (liaisons de télécommunications) et de nœuds (commutateurs, routeurs...), mis en place de telle sorte que des messages puissent être transmis d'un bout à l'autre du réseau au travers des multiples liaisons.*

*Les liaisons d'un réseau de télécommunication peuvent être réalisées grâce à des systèmes de transmission hiérarchiques.*

### Exemples de réseaux de télécommunication :

- réseau de télévision.
- réseau informatique.
- réseau de téléphonie mobile.
- réseau téléphonique commuté.
- le réseau numérique à intégration de services (RNIS).
- le réseau de téléphonie mobile public terrestre.
- le réseau Télex mondial.
- le réseau aéronautique ACARS.
- le réseau de la sécurité civile.
- le réseau Internet.

## 2.3 Le problème de conception de réseaux

En général, l'architecte réseau tient compte d'un nombre considérable de données qui lui permettent de proposer un modèle satisfaisant les conditions requises, notamment :

- Des informations sur l'ensemble des positions des sites/noeuds à interconnecter (les entreprises clientes, les abonnés au service, les noeuds d'entrée au coeur du réseau, etc).
- Des caractéristiques (le plus souvent estimées) des flux inter-sites/noeuds (volume, comportement dans le temps, etc).

Sur la base de cet ensemble de données, le processus de conception des réseaux doit spécifier la topologie du réseau ainsi que les caractéristiques des différents noeuds et connexions, le routage des trafics, etc.

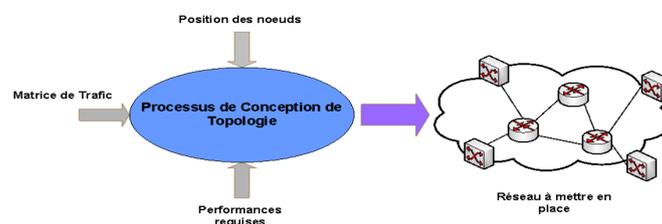


FIGURE 2.2 – processus de conception de réseaux

## 2.4 État de l'art

La conception d'un réseau est subdivisée en deux sous-problèmes différents.

1. Conception de la topologie du réseau. On part de connaissances spécifiques sur les coûts de mise en place de lignes entre différents sites du réseau. Ces coûts sont souvent indépendants du type de ligne qui sera effectivement mis en place, puisqu'ils modélisent les coûts fixes (les coûts de creuser des tranchées en cas d'installation de fibre optique, de mettre en service la fibre, etc). Une bonne partie du budget total de mise en place du réseau est affectée à cette phase.
2. Dimensionnement des lignes appelées à relier les différents noeuds du réseau, et des équipements à mettre en place dans ces noeuds.

Cette approche à longterm a été considérée comme acceptable dans la mesure où le coût des équipements était très inférieur au coût des liens. Néanmoins, de nos jours avec le déploiement massif des technologies optiques, le coût des liens devient de moins en moins cher voire comparable au coût des équipements, d'où la nécessité d'intégrer le problème de dimensionnement dans le processus de conception de topologie.

## 2.5 La notion de couche

Nous introduisons la notion de réseau multicouche, pour ramener le modèle théorique des réseaux de télécommunication à la réalité quotidienne des opérateurs. En pratique, les tests sont menés sur des réseaux à deux couches[17]. exemple(figure 2.3).

**remarque 2.2.** *Les opérateurs de télécommunication considèrent des réseaux multi-couches. Pour fixer les idées, prenons par exemple un réseau sur 2 couches (voir figure 2.3). Les noeuds sur la couche haute (couche IP) sont des routeurs qui peuvent potentiellement tous dialoguer entre eux : le graphe associé à cette couche est donc a priori complètement maillé.*

*La couche haute est souvent appelée couche virtuelle. car en réalité, à tout lien virtuel ne correspond pas forcément un lien physique. Cette couche*

virtuelle est encapsulée dans une couche de transmission (couche OTN) dont les noeuds sont des brasseurs et les arêtes sont des systèmes WDM.

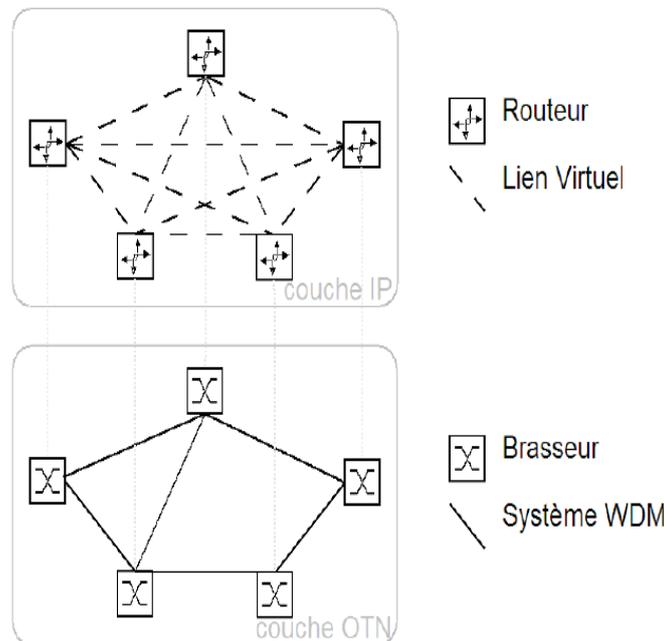


FIGURE 2.3 – exemple de réseau double couche

## 2.6 Les types de noeuds dans un réseau de télécommunication

Dans la pratique un réseau de télécommunication est composé de deux types de noeuds :

### ◆ Les noeuds terminaux :

ces noeuds se comportent comme des clients du réseau. Ce sont ces noeuds qui génèrent les demandes à acheminer à travers le réseau (exemple : les routeurs via lesquels les entreprises clientes accèdent au réseau de l'opérateur).

### ◆ Les noeuds de transit :

Ces noeuds sont utilisés comme noeuds de transit pour relier les noeuds terminaux. Un noeud de transit ne génère pas de trafic. Un tel noeud est optionnel et peut ne pas être utilisé dans le réseau.

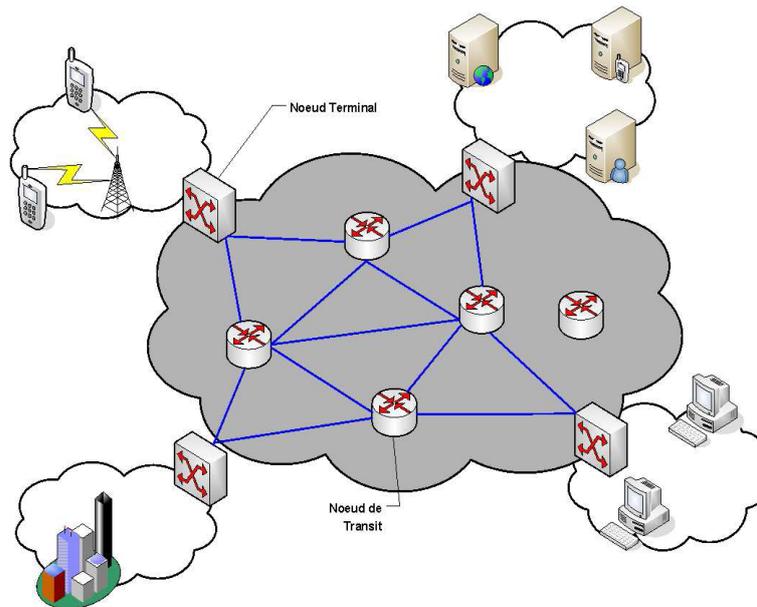


FIGURE 2.4 – Les types de noeuds dans un réseau

## 2.7 Construction de modèle

### 2.7.1 Objectif

Construire une topologie des deux couches et un dimensionnement des liens de ces couches qui rendent le réseau résistant aux pannes simples, tout en minimisant le coût global d'installation.

Sachant que les capacités des liens prennent des valeurs dans un ensemble donné de capacités et que les fonctions de coûts à considérer ne sont non seulement pas linéaires, mais même pas continues : ce sont des fonctions croissantes et constantes par morceaux.

Notre objectif est donc de déterminer un modèle, caractérisé par un problème d'optimisation combinatoire dans lequel le trafic est représenté par des multiflots.

Le réseau construit doit être résilient, c'est à dire qu'il garantit, même en cas de panne d'un équipement, qu'il existe toujours un chemin

entre deux noeuds terminaux et qu'aucune panne n'engendrera la congestion du réseau. Ceci implique des contraintes sur la connexité du graphe du réseau.

### 2.7.2 Fonctions de coûts

Dans le domaine de télécommunications, les fonctions donnant le coût d'installation d'une capacité sur un lien sont discontinues. En effet, installer une capacité sur un lien c'est installer un module sur le lien. Pour un réseau IP, les modules sont des paires de cartes que l'on installe sur les routeurs aux deux extrémités du lien, alors que pour un réseau de transmission optique, un module peut être un système WDM.

En général, on dispose de plusieurs modules différents qui sont compatibles avec le lien  $e \in E$  considéré ( $E$  est le nombre de liens que comporte le réseau). L'ensemble  $V_e$  des valeurs de capacité que l'on peut installer sur ce lien  $e$  est donc un ensemble fini de valeurs discrètes. Notons  $p(e)$  son cardinal. À chacune de ces valeurs de capacité est associé un coût d'installation sur le lien  $e$ . Notons  $\Gamma_e$  l'ensemble de ces coûts.

$V_e = \{v_e^1, \dots, v_e^{p(e)}\}$  avec  $v_e^1 < \dots < v_e^{p(e)}$  et  $\Gamma_e = \{\gamma_e^1, \dots, \gamma_e^{p(e)}\}$

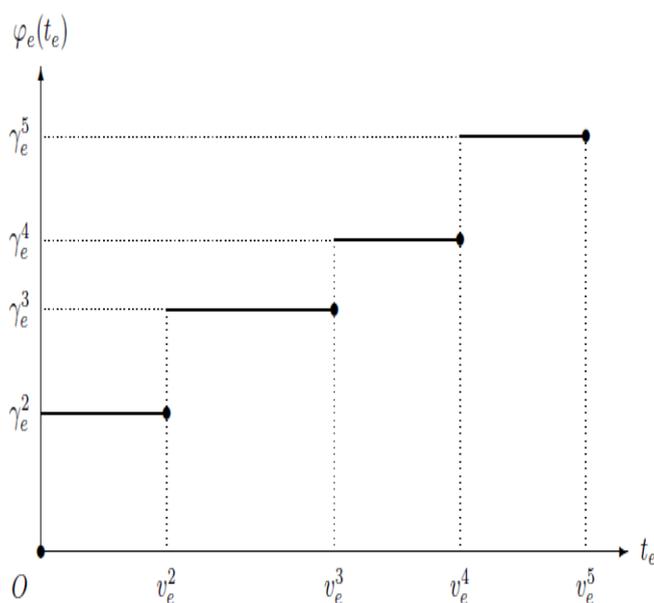


FIGURE 2.5 – exemple fonction coût

Si l'on note  $\forall e \in E$ ,  $t_e$  la quantité de trafic empruntant le lien  $e$  et  $\varphi_e(t_e)$  le coût d'installation de la capacité  $x_e$  nécessaire pour faire passer la quantité  $t_e \leq x_e$  par le lien  $e$ , alors la fonction  $\varphi_e$  est une fonction discontinue, croissante et constante par morceaux comme le présente la figure (2.5). Formellement, nous pouvons écrire :

$\forall e \in E, \forall t_e \leq v_e^{p(e)}, \exists \alpha \in \mathbb{N}$  avec  $1 \leq \alpha \leq p(e)$  tel que :

$$\begin{cases} x_e = v_e^\alpha \\ \varphi_e(t_e) = \varphi_e(x_e) = \gamma_e^\alpha \end{cases}$$

### 2.7.3 Formulation arcs-chemins

Nous avons déjà dit que les demandes de trafic sont représentées par un graphe  $G_0 = (N, E_0)$ . Si l'on note  $m_0 = \text{card}(E_0)$ , nous disposons d'un ensemble de  $m_0$  demandes. Intuitivement, on se rend bien compte que le trafic de chacune de ces demandes s'écoule de sa source jusqu'à son puits en empruntant un ou plusieurs chemins.

Afin d'obtenir la formulation naturelle de notre problème, nous modélisons l'écoulement du trafic par un multiflot réel multirouté dans le graphe  $G$ . A chaque demande indexée par  $k$  (avec  $1 \leq k \leq m_0$ ) est associé un flot de source  $s_k$ , de puits  $t_k$  et de valeur  $d_k$ , la valeur de la demande. Notons alors  $q_k$  le nombre de chemins élémentaires entre  $s_k$  et  $t_k$  dans le graphe  $G$  et  $(P_j^k)_e \in E$  le vecteur caractéristique du  $j$ -ème chemin dont les composantes  $P_j^k(e)$  ont pour valeur 1 si l'arête  $e$  est empruntée par le chemin, 0 sinon. Notons enfin  $f_j^k$  la valeur du flot  $k$  qui emprunte le  $j$ -ème chemin.

Le Problème de Dimensionnement de Réseau (PDR) s'écrit alors :

$$(PDR) \left\{ \begin{array}{l} \min \sum_{e \in E} \varphi_e(x_e) \\ \text{SC} \\ \sum_{k=1}^{m_0} \sum_{j=1}^{q_k} p_j^k(e) f_j^k \leq x_e \quad \forall e \in E \\ \sum_{j=1}^{q_k} f_j^k = d_k \quad \text{pour } 1 \leq k \leq m_0 \\ f_j^k \geq 0 \quad \text{pour } 1 \leq k \leq m_0 \text{ et } 1 \leq j \leq q_k \\ x_e \in V_e \quad \forall e \in E \end{array} \right.$$

La première contrainte représente les contraintes de capacité, elles imposent que la somme des quantités de flots qui sont routés sur une arête  $e$  ne doit pas dépasser sa capacité  $x_e$ . La deuxième contrainte traduit la conservation du trafic : elle représente la somme des portions du flot routant la demande  $k$  doit être égale à la valeur  $d_k$  de la demande.

**remarque 2.3.** *La formulation de (PDR) à réduit le nombre de contraintes à 2 contraintes seulement, qui est à l'origine égal à  $2|E| + |E_0|$ , qui augmente linéairement avec la densité du graphe  $G$  et avec le nombre de demandes. elle comporte aussi un nombre très élevé de variables, égal à  $\sum_k q_k$ , qui augmente exponentiellement avec le nombre d'arêtes dans  $G$ .*

*Cette remarque n'empêche pas la formulation arcs-chemins d'être très utilisée dans le domaine des télécommunications car elle permet d'ajouter facilement des contraintes de routage des demandes qui se traduisent simplement par des contraintes faisant intervenir les vecteurs caractéristiques des chemins.*

#### 2.7.4 Passage à la formulation capacité

On note  $X^*$  (le polyèdre des solutions réalisables) L'ensemble des solutions réalisables du problème de dimensionnement est appelé dominant du polyèdre des multiflots réalisables. Si on note  $x = (x_e)_{e \in E}$  un vecteur de capacité et  $f = (f_k)_{k \in K}$  un multiflot routant l'ensemble  $K$  des demandes de trafic, alors :

$X^* = \{x \in \mathbb{R}_+^{|E|} \text{ tel qu'il existe un multiflot } f \text{ compatible avec } x \}$ .

**lemme 2.1. (Farkas-Minkowski)** Soit  $A$  une matrice de dimension  $m \times n$  et  $b$  un  $m$ -vecteur. Le système des inégalités linéaires  $Ax \leq b$  a des solutions positives  $x \geq 0$  si et seulement si :

$$\forall y \in \mathbb{R}_+^m \text{ on a } y.A \geq 0 \Rightarrow y.b \geq 0$$

La caractérisation du polyèdre  $X^*$  est obtenue en appliquant le lemme de ( Farkas-Minkowski ) à la formulation arcs-chemins du problème d'existence d'un multiflot compatible. On a alors :

$$X^* = \left\{ x \in \mathbb{R}_+^{|E|} \text{ tel que } \forall \mu \in \mathbb{R}_+^{|E|}; \sum_{(i,j) \in E} \mu_{ij} x_{ij} \geq \sum_{(i,j) \in E_0} l_{ij}^*(\mu) d_{ij} \right\}$$

ou  $l_{ij}^*(\mu)$  est la longueur du plus court chemin entre  $i$  et  $j$  dans le graphe  $G$  dans lequel toute arête  $(i, j) \in E$  est munie de la longueur  $\mu_{ij}$ .

**remarque 2.4.** La caractérisation du polyèdre  $X^*$  pose toujours problème. quelque soit l'algorithme utilisé pour calculer les plus courts chemins dans un graphe, la recherche d'une inégalité caractéristique de  $X^*$  peut devenir rapidement rédhibitoire, pour un vecteur  $\mu$ , il faudrait calculer les plus courts chemins pour tout couple  $(i, j) \in E_0$ .

**Définition 2.2.** On note par  $Met_n$  le cône métrique d'ordre  $n$ ,  $n$  désignant le nombre de sommets de  $G$  tel que :

$$Met_n = \left\{ \lambda \in \mathbb{R}_+^{\frac{n(n-1)}{2}} \text{ tel que } \lambda_{ij} \leq \lambda_{ik} + \lambda_{kj} \text{ avec } \begin{cases} 1 \leq i \leq j \leq n \\ k \neq i, k \neq j. \end{cases} \right\}$$

Nous considérons un nouvel ensemble  $X$  où  $\mu$  et les longueurs des plus courts chemins ont disparu, mais où  $Met_n$  apparait :

$$X = \left\{ x \in \mathbb{R}_+^{|E|} \text{ tel que } \forall \lambda \in Met_n \text{ on a } \sum_{(i,j) \in E} \lambda_{ij} x_{ij} \geq \sum_{(i,j) \in E_0} \lambda_{ij}(\mu) d_{ij} \right\}$$

**Proposition 2.1.** Les deux ensembles  $X^*$  et  $X$  sont identiques :  $X^* = X$ .

**Théorème 2.1. (*dit japonais*)**

Le couple  $(x, d)$  est réalisable pour le problème de dimensionnement si et seulement si  $\forall \lambda \in \text{Met}_n, (x - d)^T \cdot \lambda \geq 0$ . Ces dernières inégalités sont appelées inégalités métriques.

Avec cette nouvelle caractérisation, une inégalité métrique sera générée en choisissant un  $\lambda \in \text{Met}_n$ , ce qui se fait en résolvant un programme linéaire continu intégrant les inégalités caractéristiques de  $\text{Met}_n$ . Ces résultats nous permettent d'introduire la formulation capacité du Problème de Dimensionnement de Réseau :

$$(PDR) \begin{cases} \min \sum_{e \in E} \varphi_e(x_e) \\ sc \\ x \in X \\ x_e \in V_e \end{cases} \quad \forall e \in E$$

**remarque 2.5.** Le (PDR) est exprimé ici par un programme linéaire généralisé. La contrainte " $x \in X$ " cache en réalité une infinité d'inégalités métriques données par une infinité de  $\lambda \in \text{Met}_n$ .

**résultat 2.1. (H. MINKOWSKI)**

Le cône métrique peut être généré à partir de ses rayons extrêmes, c'est-à-dire à partir des  $\lambda \in \text{Met}_n$  qui ne peuvent pas s'exprimer comme des combinaisons linéaires d'autres éléments de  $\text{Met}_n$ .

Le nombre de rayons extrêmes de  $\text{Met}_n$ , dépendant de  $n$ , est fini mais reste grand (plus d'un milliard pour  $\text{Met}_8$ ).

Les inégalités métriques formées à partir des éléments de  $\text{Ray}_n$ , l'ensemble des rayons extrêmes de  $\text{Met}_n$ , est nécessaire et suffisant pour caractériser entièrement le dominant du polyèdre des multiflots réalisables. Le problème de dimensionnement, dans sa formulation capacité, comporte donc  $|E|$  variables et  $|\text{Ray}_n|$  contraintes.

### 2.7.5 Introduction de couche

Nous introduisons la notion de réseau multicouche dans le modèle théorique des réseaux de télécommunications, pour le ramener à la réalité quotidienne des opérateurs.

On note par  $L$  l'ensemble de couches, Chaque couche se modélise par un graphe support :  $\forall l \in L, G_l = (N, E_l)$

Comme précédemment,  $N$  est l'ensemble des  $n$  noeuds du réseau qui sont identiques pour toutes les couches et  $E_l$  est l'ensemble des liens disponibles pour la couche  $l$ . Notons  $m_l$  le cardinal de cet ensemble.

On définit la fonction de coûts  $tq$  : l'ensemble  $V_e$  des  $p(e)$  valeurs possibles pour la capacité du lien  $e \in E_l$ , l'ensemble  $\Gamma_e$  des coûts associés à chacune de ces valeurs, et enfin nous notons  $\varphi_e$  la fonction de coûts qui en résulte.

Avant de lancer la version multicouche du problème de dimensionnement de réseau, nous notons à partir de maintenant :

$x_l(e)$  la capacité installée sur le lien  $e$  de la couche  $l$ . Nous disposons donc, pour chaque couche  $l$ , du vecteur capacité de la couche :  $x_l = (x_l(e))_{e \in E_l}$ . Pour des raisons esthétiques, notons  $x_0 = d = (d_{ij})_{(i,j) \in E_0}$ .

Notre but est de dimensionner simultanément toutes les couches que comporte le réseau, Il faut donc trouver simultanément un multiflot par couche qui soit compatible avec les capacités installées sur les liens de cette couche. Ce multiflot doit également permettre de router l'ensemble des demandes de la couche considérée, qui sont en fait les capacités de la couche juste au dessus .

$$\forall l \in L, \\ X = \left\{ x \in \mathbb{R}_+^{m_l} \text{ tel que } \forall \lambda \in \text{Ray}_n \text{ on a } \sum_{e=(i,j) \in E_l} \lambda_{ij} x(e) \geq \sum_{e=(i,j) \in E_{l-1}} \lambda_{ij} x_{l-1}(e) \right\}$$

D'où la formulation capacité du Problème de Dimensionnement de Réseau Multicouche :

$$(PDRM) \left\{ \begin{array}{l} \min \sum_{l \in L} \sum_{e \in E_l} \varphi_e(x_l(e)) \\ sc \\ x_l \in X_l \quad \forall l \in L \\ x_l(e) \in V_e \quad \forall l \in L, \forall e \in E_l \end{array} \right.$$

### 2.7.6 La sécurisation

Dans le but d'adapter le modèle à la réalité, il faut sécuriser le réseau que l'on veut créer, c'est-à-dire, le rendre résistant aux pannes d'un noeud ou d'un lien. Dans ce document, nous ne nous intéressons qu'aux pannes des liens du réseau.

En effet le réseau garantit même en cas de panne d'un équipement, qu'il existe toujours un chemin entre deux noeuds terminaux et qu'aucune panne n'engendrera la congestion du réseau. Ceci implique des contraintes sur la connexité du graphe du réseau.

Nous introduisons maintenant le nouveau vecteur capacité  $x_l^{\tilde{e}} \in \mathbb{R}_+^{m_l}$  par couche, qui est modifié de façon à simuler la panne de l'arête  $\tilde{e}$  de la couche  $l$ . On a alors :

$$\forall l \in L, \forall e \in E_l, x_l^{\tilde{e}}(e) = \begin{cases} 0 & \text{si } e = \tilde{e}, \\ x_l(e) & \text{sinon.} \end{cases}$$

Nous introduisons également de nouveaux polyèdre des multiflots réalisables qui prennent en compte la panne de l'arête :  $\forall l \in L, \forall \tilde{e} \in E_l$

$$X_l^{\tilde{e}} = \left\{ x \in \mathbb{R}_+^{m_l} \text{ tel que } \forall \lambda \in \text{Ray}_n \quad \sum_{e=(i,j) \in E_l - \{\tilde{e}\}} \lambda_{ij} x(e) \geq \sum_{e=(i,j) \in E_{l-1}} \lambda_{ij} x_{l-1}(e) \right\}$$

On déduit alors la formulation capacité du problème de dimensionnement de réseau multicouche sécurisé que l'on peut écrire comme suit :

$$(PDRMS) \left\{ \begin{array}{l} \min \sum_{l \in L} \sum_{e \in E_l} \varphi_e(x_l(e)) \\ s.c. \\ x_l^{\tilde{e}} \in X_l^{\tilde{e}} \\ x_l(e) \in V_e \end{array} \quad \begin{array}{l} \forall l \in L, \forall \tilde{e} \in E_l \\ \forall l \in L, \forall e \in E_l \end{array} \right.$$

### 2.7.7 Linéarisation du problème

On a  $V_e$  représente l'ensemble des valeurs possibles (rangées dans l'ordre croissant) de la capacité du lien  $e$  et  $\Gamma_e$  l'ensemble des coûts d'installation associés à chacune de ces valeurs, tels que :

$$V_e = \{ v_e^1, \dots, v_e^{p(e)} \} \quad \text{et} \quad \Gamma_e = \{ \gamma_e^1, \dots, \gamma_e^{p(e)} \}$$

nous introduisons des variables bivalentes  $x_l^\alpha(e)$ ,  $\alpha = 2, \dots, p(e)$  associées à chaque marche de la fonction de coûts en escalier. La variable  $x_l^\alpha(e)$  vaut 1 si et seulement si le module permettant de passer de la capacité  $v_e^{\alpha-1}$  à la capacité  $v_e^\alpha$  pour le lien  $e$  de la couche  $l$  est effectivement installé. On a donc :

$$\forall \alpha = 1, \dots, p(e) \quad x_l(l) = v_e^\alpha \Leftrightarrow \begin{cases} x_l^\beta(e) = 1 & \forall \beta \leq \alpha \\ x_l^\beta(e) = 0 & \forall \beta > \alpha \end{cases}$$

Et nous pouvons écrire :

$$x_l(e) = \sum_{\alpha=2}^{p(e)} (v_e^\alpha - v_e^{\alpha-1}) x_l^\alpha(e)$$

$$\varphi_e(x_l(e)) = \sum_{\alpha=2}^{p(e)} (\gamma_e^\alpha - \gamma_e^{\alpha-1}) x_l^\alpha(e)$$

De même, pour la contrainte d'intégrité :

$$x_l(e) \in V_e \Leftrightarrow \begin{cases} x_l^\alpha(e) \in \{0, 1\} & \text{pour } \alpha = 2, \dots, p(e) \\ x_l^\alpha(e) \leq x_l^{\alpha-1}(e) & \text{pour } \alpha = 3, \dots, p(e) \end{cases}$$

On déduit alors la formulation capacité du problème de dimensionnement de réseau multicouche sécurisé, qu'on peut le représenter sous forme d'un programme linéaire en variables bivalentes :



# Conclusion

*Nous avons étudié dans ce mémoire le problème de flot et de multiflot dans les réseaux, sachant qu'ils sont fondamentaux et très souvent utiles.*

*Lors de la conception d'un réseau, l'opérateur se base sur une estimation des demandes en trafic dans le futur réseau concernant des échelles de temps relativement longues. Ces estimations sont généralement faites grâce à des études de marché. Dans la pratique, ces estimations ne sont pas assez précises. Durant la phase d'exploitation du réseau, l'opérateur doit anticiper l'évolution globale du trafic.*

*Nous avons traité le problème de conception de réseaux. Ce problème se pose généralement aux opérateurs qui souhaitent mettre en place une nouvelle infrastructure de télécommunication. La résolution de ce problème représente une tâche très complexe pour un opérateur, et en règle générale, fort coûteuse étant donné l'importance des investissements nécessaires.*

*Nous nous intéresserons ensuite aux différentes formulations de flot afin de se ramener à un modèle mathématique, qui généralise le problème.*

# Bibliographie

- [1] *SLIMI L, Problème du flot dans les réseaux de transport et application sur le réseau hydraulique de la ville de Tizi ousou, UMMTO, Algerie, 2012.*
- [2] *GEORGIOS P, dimensionnement robuste des réseaux de télécommunications face à l'incertitude de la demande, Université Paris 1, France, 2009.*
- [3] *CLAUDE B, Graphes et hypergraphes, Dunod, 1970.*
- [4] *JACQUES L, Théorie des graphes, Canada, 1981.*
- [5] *MATHIEU T, Modélisation et optimisation du déploiement des réseaux de télécommunications, Laboratoire Cédric, France, 2009.*
- [6] *MICHEL S, Optimisation combinatoire, Hermann, France, 1984.*
- [7] *YANNICK P, Diagnostic décentralisé de systèmes à événements discrets : application aux réseaux de télécommunications, Université Rennes 1, France, 2002.*
- [8] *DAVID C, Algorithmique et optimisation de réseaux de communications optiques, Université Nice Sophia Antipolis, France, 2001.*
- [9] *CÉDRIC B, Résolution exacte et approchée de problèmes de multi-flot entier et de multicoupe, 2006.*
- [10] *ZIED B, Conception et optimisation robuste des réseaux de télécommunications, Université Toulouse - Paul Sabatier, France, 2010.*
- [11] *CLAUDE P, Optimisation convexe dans les réseaux, avec applications au trafic routier et à l'énergie électrique, Ecole polytechnique fédérale de lausanne, Suisse, 1987.*
- [12] *JEROME T, Conception de réseaux haut débit sous contrainte de sécurisation, Université blaise pascal - clermont-ferrand II, France, 2007.*

[13] *BENOÎT M, Méthode exacte pour résoudre le problème de dimensionnement multicouche de réseau sécurisé, Institut d'informatique d'entreprise, France.*

[14] *MICHEL M, Synthèse optimale d'un réseau de télécommunication avec contraintes de sécurité, 1981.*

[15] *MICHEL M. et Michel G., Graphes et algorithmes, Lavoisier, 2009.*