INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Un algorithme d'allocation de bande passante satellitaire

Sara Alouf — Eitan Altman — Jérôme Galtier — Jean-François Lalande — Corinne Touati

N° 5172

Avril 2004

THÈME 1

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--5172--FR

Un algorithme d'allocation de bande passante satellitaire*

Sara Alouf, Eitan Altman, Jérôme Galtier†, Jean-François Lalande, Corinne Touati

Thème 1 — Réseaux et systèmes
Projets Maestro et Mascotte

Rapport de recherche n° 5172 — Avril 2004 — 37 pages

Résumé : Ce rapport présente un algorithme d'allocation de ressources pour les réseaux satellitaires. Il s'agit de prévoir un plan d'allocation en temps/fréquence pour un ensemble de terminaux ayant une configuration géométrique définie et soumis à des contraintes d'interférence. On cherche à minimiser la taille du plan de fréquences tout en garantissant que toutes les demandes des terminaux, en termes de bande passante et pour différents types, sont satisfaites. L'algorithme proposé repose sur deux techniques principales: la génération de configurations admissibles pour les contraintes d'interférence par des heuristiques, la programmation mixte linéaire/entière utilisant la génération de colonnes. La solution obtenue permet de prévoir un plan d'allocation admissible avec des garanties d'optimalité et permet aussi de mettre en évidence les configurations d'interférences qui entravent la génération de bonnes solutions.

Mots-clés : Allocation de ressources, plan de fréquences, programmation linéaire mixte

* Cette étude a été réalisée dans le cadre du contrat 1 02 E 0306 00 41620 01 2 avec Alcatel Space industries, (Convention de Recherche A 56918)

† France Telecom R&D

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

An algorithm for satellite bandwidth allocation

Abstract: This report presents an allocation algorithm for resources in satellite networks. It deals with planning a time/frequency plan for a set of terminals with a known geometric configuration and with interference constraints. The objective is to minimize the size of the frequency plan with the guaranty that the different types of demands are satisfied, each type using different bandwidth. The proposed algorithm uses two main technics: the first generates admissible configurations for the interference constraints, the second uses mixed linear/integer programming with column generation. The obtained solution estimates a possible allocation plan with optimality guarantees and highlights the frequency interferences which degrade the construction of good solutions.

Key-words: Resources allocation, frequency plan, mixed linear programming

Table des matières

Chapitre 1

Modélisation

Le problème se résume comme suit. Un satellite de communication doit établir sa plage de réception d'un parc de terminaux répartis sur sa zone de couverture. Du point de vue du canal radio, le satellite divise le spectre en fréquence et en espace. Géographiquement, les terminaux sont répartis sur des *spots*, qui correspondent à des équipements de réception du satellite.

1.1 Réutilisation spatiale

Le système tire avant tout parti d'une *réutilisation spatiale* de la bande passante dans la mesure où le taux d'interférence est acceptable. Cette réutilisation spatiale est possible de par la forme hexagonale des spots prévue par le modèle, très proche de celui de [3]. Pour ce faire, on utilise une première division de la bande de fréquences en zones de tailles égales appelées *couleurs*, représentée en figure 1.1. De manière correspondante, on affecte à chaque spot une couleur unique, qui fixe la zone exclusive de couleurs dans laquelle il pourra communiquer. Cette affectation de couleurs est faite de sorte que deux spots (trop) voisins ne puissent émettre dans la même plan de fréquences, sans quoi l'interférence générée serait inacceptable.

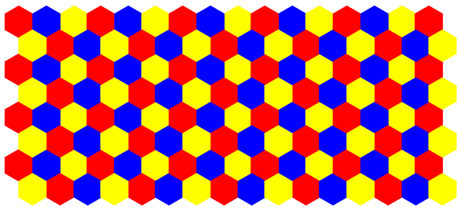

FIG. 1.1 – Configuration spatiale des spots

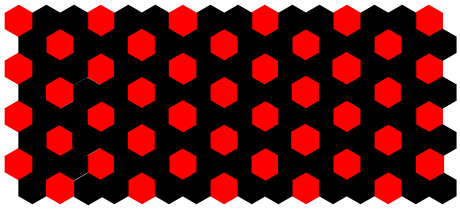

FIG. 1.2 – Configuration spatiale pour une unique couleur

Ainsi, au sein de chacune des différentes couleurs se définit un problème complètement indépendant de celui posé par les autres couleurs. Par conséquent, nous considérerons par la suite uniquement le probème d'une unique couleur, c'est-à-dire sur les spots ayant cette même couleur et la bande de fréquence correspondante. On notera N ce nombre de spots, et B la taille de la bande passante. En pratique, on s'intéresse à $N \leq 32$. Cette configuration est représentée en figure 1.2.

La bande de fréquence est potentiellement réutilisée d'un spot à l'autre. Cette réutilisation n'est cependant possible que si le niveau d'interférence généré par l'ensemble des spots utilisant simultanément la ressource radio est acceptable.

1.2 Niveau d'interférence

Pour tenir compte des conditions réelles de la propagation radio, il est nécessaire de tenir compte de l'emplacement des terminaux au sein même d'un spot. Pour cela on introduit une notion de subdivision géographique d'un spot en *zones*. Chaque terminal est ainsi complètement caractérisé géographiquement par sa zone. En pratique, un spot a deux ou trois zones.

Chaque zone génère une certaine interférence sur tous les autres spots $I(z, s)$ (si s appartient au spot z on note $I(s, z) = 0$). Chaque zone a aussi un gain $G(z)$. Il faut absolument que sur tout canal utilisé, le gain divisé par la somme des interférences soit supérieur à un certain seuil σ, autrement dit, si $spot(z)$ est le spot contenant z, pour tout instant t et fréquence f, z peut être active en (t, f) seulement si

$$\frac{G(z)}{\sum_{z' \text{ active en } (t,f)} I(spot(z), z')} \geq \sigma \tag{1.1}$$

Dans les tests effectués par la suite, la valeur du gain en décibels est prise aléatoirement entre deux valeurs bornes. Pour l'interférence, on a distingué l'interférence de voisinage, générée sur les (six) spots les plus proches, et l'interférence globale, générée sur l'ensemble des spots, toutes deux étant prises aussi aléatoirement en dB dans leurs intervalles respectifs. La valeur finale de l'interférence est un barycentre à poids fixe des deux valeurs ci-dessus.

Type	Nombre maximal de porteuses sur la bande	Nombre maximal de time-slots par trame
1	192	18
2	96	36
3	12	288
4	6	576

TAB. 1.1 – Valeurs test des types de terminaux.

1.3 Types de terminaux et demande

Les terminaux ont des capacités variables d'émission. Ces capacités sont indiquées par la notion de type. Un type correspond à la bande de fréquence qu'utilise un terminal. Le rapport entre les tailles de bande passante de deux types différents est un entier ou l'inverse d'un entier. L'intervalle de temps effectif d'un time-slot dépend du type utilisé, de sorte que le volume de données transmises lors d'un time-slot est une constante du problème. De manière équivalente, le produit largeur de bande d'une porteuse (la porteuse représente la bande de fréquence utilisée par le terminal) multiplié par durée d'un time-slot, notée Δ, est constant pour tous les types. Pour mémoire, les valeurs test du problème sont indiquées dans la table 1.1.

Chaque spot est divisé en zones, et chaque zone a une demande spécifique à satisfaire, exprimée en nombre de time-slots (ou unités de base de données) par types de terminaux.

$$\forall z_i, \forall k,\ d_{z_i}^{t_k} = \text{demande de } z_i \text{ en type } t_k$$

Un type de terminal t_k se caractérise par sa largeur de bande t_k^b et son temps de transmission t_k^t. On a, pour tout type k :

$$\Delta = t_k^b t_k^t$$

1.4 Etude d'un cas simple

Pour bien comprendre la modélisation, on peut imaginer qu'il n'y a qu'un seul type de terminal. Le canal radio se découpe alors en "time-slots" - comme en GSM - où un certain nombre de zones peuvent émettre simultanément. La première question est de savoir : quelles zones peuvent alors émettre dans un time-slot et à une fréquence donnés ?

Prenons un exemple simple, illustré par la figure 1.3. Sur cette figure, chaque spot est découpé en deux zones qui ont une intérférence spécifique sur les autres spots. Par exemple, on peut prendre le tableau numérique 1.2, qui précise, pour chaque zone, son gain ainsi que les interférences qu'elle génère sur les autres spots. Concentrons-nous maintenant sur la zone 1.1. Dans quelle mesure cette zone peut-elle émettre ? Cela dépend des états des zones qui interfèrent. Suivant que les zones 2.1, 2.2, 3.1 et 3.2 sont allumées ou pas, son rapport C/I va évoluer. La table 1.3 résume les 9 cas de figure qui peuvent se présenter.

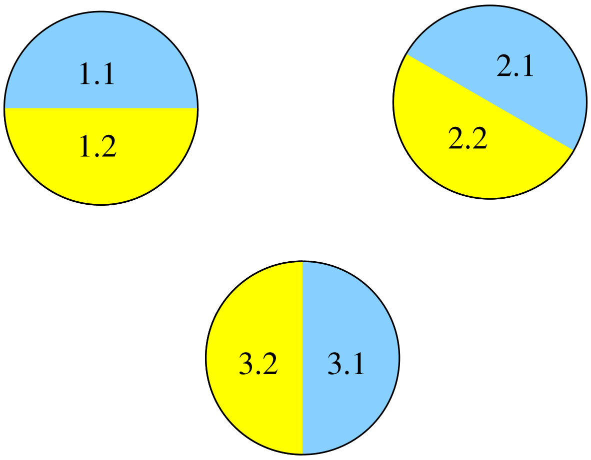

FIG. 1.3 – Petit exemple sur 3 spots.

Zone	Gain	I(.,Spot 1)	I(.,Spot 2)	I(.,Spot 3)
1.1	4	-	5	3
1.2	6	-	5	7
2.1	3	4	-	2
2.2	8	7	-	10
3.1	5	3	7	-
3.2	5	7	3	-

TAB. 1.2 – Tableau des gains et interférences des zones de l'exemple.

C/I (Zone 1.1)	Zone 2.1 allumée	Zone 2.2 allumée	Spot 2 éteint
Zone 3.1 allumée	0.57	0.40	1.33
Zone 3.2 allumée	0.36	0.29	0.57
Spot 3 éteint	1.00	0.57	-
C/I (Zone 1.2)	**Zone 2.1 allumée**	**Zone 2.2 allumée**	**Spot 2 éteint**
Zone 3.1 allumée	0.86	0.60	2.00
Zone 3.2 allumée	0.55	0.43	0.86
Spot 3 éteint	1.50	0.86	-
C/I (Zone 2.1)	**Zone 1.1 allumée**	**Zone 1.2 allumée**	**Spot 1 éteint**
Zone 3.1 allumée	0.25	0.25	0.43
Zone 3.2 allumée	0.38	0.38	1.00
Spot 3 éteint	0.60	0.60	-
C/I (Zone 2.2)	**Zone 1.1 allumée**	**Zone 1.2 allumée**	**Spot 1 éteint**
Zone 3.1 allumée	0.67	0.67	1.14
Zone 3.2 allumée	1.00	1.00	2.67
Spot 3 éteint	1.60	1.60	-
C/I (Zone 3.1)	**Zone 1.1 allumée**	**Zone 1.2 allumée**	**Spot 1 éteint**
Zone 2.1 allumée	1.00	0.56	2.50
Zone 2.2 allumée	0.38	0.29	0.50
Spot 2 éteint	1.67	0.71	-
C/I (Zone 3.2)	**Zone 1.1 allumée**	**Zone 1.2 allumée**	**Spot 1 éteint**
Zone 2.1 allumée	1.00	0.56	2.50
Zone 2.2 allumée	0.38	0.29	0.50
Spot 2 éteint	1.67	0.71	-

TAB. 1.3 – Détail des cas d'interférence possibles pour différentes zones.

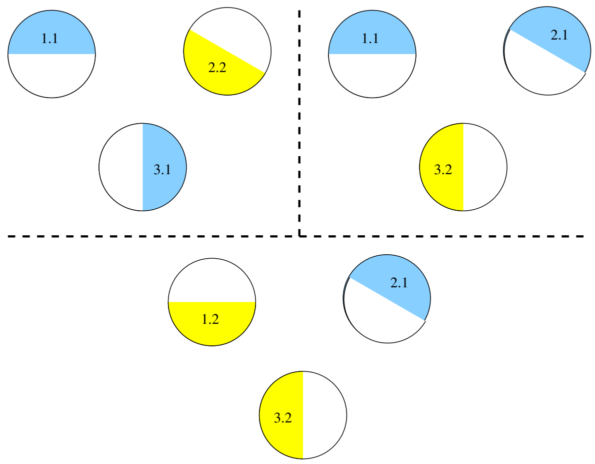

FIG. 1.4 – Ensemble des configurations d'émission possibles pour un seuil à 0.30.

En fixant le seuil d'admissibilité à 0.30, toutes les combinaisons possibles de spots avec deux zones allumées dans deux spots différents sont possibles, et trois combinaisons avec trois spots actifs sont possibles. Nous les montrons sur la figure 1.4. Le lecteur pourra vérifier que ces familles vérifient les conditions d'interférence pour chaque zone et que les autres configurations à trois spots actifs ne les vérifient pas.

Du point de vue de l'efficacité du système, il est clair que les configurations à trois zones allumées sont plus intéressantes. En effet, elles permettent de faire fonctionner simultanément trois spots, alors que les autres configurations ne permettent que deux fonctionnements en parallèle. Par la suite, nous appellerons les trois configurations de la figure 1.4 les familles 1, 2 et 3.

Ainsi, si la demande est fixée à 100 time-slots par zone de manière uniforme, il est clair que chaque spot doit recevoir 200 time-slots et que, dans l'ensemble, il faudra au minimum 200 time-slots pour écouler cette demande. Or ce minimum peut être atteint. En effet, il suffit que dans les 100 premiers time-slots les zones 1.1, 2.2 et 3.1 émettent simultanément, puis dans les 100 suivants, les zones 1.2, 2.1 et 3.2.

1.5 Etude d'un cas plus complexe

Prenons maintenant la demande un peu plus complexe du tableau 1.4. On voit que dans cet exemple, chaque spot doit recevoir 200 time-slots. Mais il faut plus que 200 time-slots pour écouler cette demande, car on ne peut pas utiliser de manière efficace les trois configurations citées plus haut en permanence ! Une méthode néanmoins efficace peut consister à utiliser dans un premier temps 50 time-slots dédiés aux zones 1.1, 2.2 et 3.1, puis 50 aux zones 1.2, 2.1 et 3.2, puis 50 aux zones 1.2 et 2.2, puis 50 aux zones 1.2 et 3.1, puis 50

Zone	Demande (time-slots)
1.1	50
1.2	150
2.1	50
2.2	150
3.1	150
3.2	50

TAB. 1.4 – Tableau des demandes de l'exemple.

Nombre de time-slots	Configuration (famille) utilisée
50	Zone 1.1 + Zone 2.2 + Zone 3.1
50	Zone 1.2 + Zone 2.1 + Zone 3.2
50	Zone 1.2 + Zone 2.2
50	Zone 1.2 + Zone 3.1
50	Zone 2.2 + Zone 3.1

TAB. 1.5 – Solution de l'exemple.

Nombre de time-slots	Configuration (famille) utilisée
100	Zone 1.1 + Zone 2.2 + Zone 3.1
50	Spot 1 + Zone 2.1 + Zone 3.2
50	Zone 1.2 + Zone 2.2
50	Zone 1.2 + Zone 3.1

TAB. 1.6 – Une description plus efficace de la solution de l'exemple.

aux zones 2.2 et 3.1. En tout, c'est donc 250 time-slots qui seront utilisés pour écouler la demande. Cette solution peut se synthétiser comme expliqué sur le tableau 1.5, où on associe un nombre de time-slots à des configurations. Dans la suite du document, on associera le terme “famille” à ces configurations admissibles, et on déterminera le nombre de time-slots où ces familles seront utilisées.

Pour une meilleure résolution du problème, d'autres notions pourront être utiles. Par exemple, notons que les familles 2 et 3 diffèrent seulement par le spot 1. En fait, il est correct de dire qu'il est possible d'allumer simultanément *tout* le spot 1 avec les zones 2.1 et 3.2. Ainsi, nous allons créer, dans des conditions mathématiques qui seront précisées dans la suite, des configurations qui en résument plusieurs autres en laissant de la flexibilité sur la zone que l'on allume au sein d'un spot. Il est également possible, pour décrire une solution du problème, d'utiliser une famille qui a “trop” de zones allumées, dans le sens où certaines resteront inutilisées pendant l'utilisation effective du système. Ainsi, on pourrait décrire la solution du problème de manière plus efficace par le tableau 1.6.

Chapitre 2

Résolution du problème

2.1 Résolution du problème d'interférence

La solution au problème des questions d'interférence vient de la constatation simple que pour toute ressource radio, et donc pour tout couple temps-fréquence (t, f), il existe une famille Z de zones simultanément actives, qui vérifient donc :

$$\forall z \in Z \quad \frac{G(z)}{\sum_{z' \in Z} I(spot(z), z')} \geq \sigma.$$

Il est alors assez naturel d'utiliser ce concept d'émissions non concurrentes de manière similaire à ce qui ce fait pour le coloriage de graphes [5]. Dans le cas du coloriage de graphe, on s'appuie sur un ensemble de familles de sommets indépendants. Ici, on utilise des familles de zones pouvant communiquer simultanément.

Le nombre de familles vérifiant ce critère est en pratique très grand pour que l'on puisse les décrire de manière exhaustive. Par ailleurs, il sera difficile de bâtir une solution uniquement en se basant sur ce concept élémentaire de famille. Par conséquent nous avons introduit la notion de super-zone au sein d'un spot. Une super-zone contient plusieurs zones d'un même spot. Si une famille contient une super-zone, cela indique que chacune des zones qu'elle contient peut émettre avec les autres zones en respectant les conditions d'interférence. Les paramètres d'une super-zone ζ regroupant deux zones z et z' d'un même spot sont les suivants :

$$\begin{aligned} G(\zeta) &= \min(G(z), G(z')) \\ I(s, \zeta) &= \max(I(s, z), I(s, z')) \quad \text{pour tout spot } s \end{aligned}$$

La méthode de résolution consiste donc, dans un premier temps, à générer des familles F_i constituées de zones ou super-zones de spots distincts.

2.1.1 Génération de familles génériques

Le seuil d'interférence σ est une donnée du problème. Si σ est très faible (par exemple 10dB, ce qui n'est pas très réaliste), tous les spots peuvent être actifs en (t, f). Plus le seuil est élevé, moins de spots peuvent être actifs simultanément et dans la même bande de fréquence.

A ce stade, il est judicieux de donner de plus amples détails sur les signaux interférents présents dans (1.1). Plus une antenne est directive, plus petits sont les lobes secondaires, autrement dit plus faible est la puissance du signal interférent. N'ayant pas à notre portée de simulation du niveau du signal interférent dans toutes les zones géographiques, nous adoptons le raisonnement suivant : les zones voisines sont celles qui génèrent le plus d'interférence l'une sur l'autre ; toutefois, les zones lointaines génèrent également des interférences, quoique d'intensité plus faible. Nous distinguons ainsi l'interférence de voisinage, générée sur les (six) spots les plus proches, et l'interférence globale, générée par l'ensemble des spots. Dans les tests effectués, les puissances des signaux interférents sont prises aléatoirement entre deux valeurs bornes (11db et 15db), indépendamment de la distance séparant le terminal interférent du terminal émettant le signal utile. Pour pallier à ceci, nous adoptons la méthode suivante pour calculer le niveau global des signaux interférents :

1. calculer $S_1(z) = \sum_{z' \text{ active dans spot voisin}} I(spot(z), z')$, la somme des interférences générées sur la zone z par les zones actives en (t, f) dans les spots voisins ;
2. calculer $S_2(z) = \sum_{z' \text{ active}} I(spot(z), z')$, la somme des interférences générées sur la zone z par toutes les zones actives en (t, f), y compris celles dans les spots voisins ;
3. calculer $I(z)$, la valeur finale de l'interférence générée sur la zone z, comme étant un barycentre à poids fixe des deux valeurs ci-dessus : $I(z) = \gamma S_1(z) + (1 - \gamma) S_2(z) = S_1(z) + (1 - \gamma)(S_2(z) - S_1(z))$.

Ainsi, les signaux interférents générés par les spots lointains ont été attenués d'un facteur $1 - \gamma$ (dans les tests $\gamma = 0.8$). À noter qu'il suffit de prendre $\gamma = 0$ pour obtenir l'interférence générée par l'ensemble des spots indépendamment de leurs positions géographiques, alors que si $\gamma = 1$ nous obtenons l'interférence de voisinage. Le critère d'allocation (1.1) devient alors

$$\frac{G(z)}{I(z)} \geq \sigma \tag{2.1}$$

Revenons à notre problème qui consiste à sélectionner le nombre maximal de zones pouvant être actives en (t, f). Vu que l'interférence de voisinage est décisive dans le calcul de $I(z)$ dans (2.1), nous allons répartir géographiquement les spots qui seront éteints en (t, f). À noter qu'à part les spots à la limite de la couverture du satellite, tout spot est voisin de 6 autres spots. Notre but est donc de répartir les spots éteints de façon à éviter les situations où plusieurs spots voisins sont éteints. Tirant profit de la disposition géographique des spots, nous proposons dans un premier temps d'éteindre un spot tous les sept spots sur une même ligne horizontale, et de décaler les spots éteints dans les lignes adjacentes, de façon à ce que tout spot allumé ait exactement un spot voisin éteint. Ce schéma, représenté dans la figure 2.1, est désigné sous le code 6/7 puisque 6 spots sur 7 sont allumés, et les spots allumés forment une famille 6/7. Il y a, bien évidemment, 7 familles 6/7 distinctes.

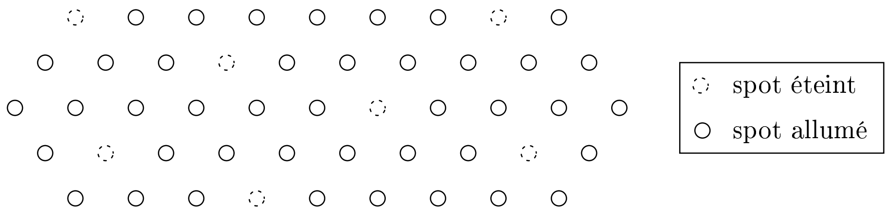

FIG. 2.1 – Illustration d'une famille 6/7

Quand le seuil d'interférence σ est plutôt élevé, il arrive que l'on ne puisse pas satisfaire le critère (2.1) pour toutes les zones allumées dans le schéma 6/7. Nous introduisons donc le schéma 5/7 dans lequel 5 spots sur 7 sont allumés, ou en d'autres termes, tout spot allumé a 2 voisins sur 6 éteints (à part les spots à la limite de la couverture du satellite qui ont au plus 2 voisins éteints). Le schéma 5/7 est représenté dans la figure 2.2. Les spots allumés forment une famille 5/7 et il y a, au total, $7 \times 3 = 21$[1] familles 5/7 distinctes.

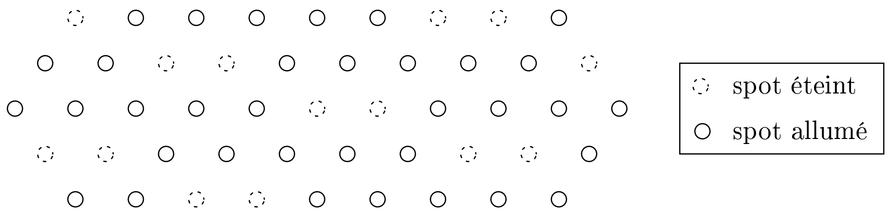

FIG. 2.2 – Illustration d'une famille 5/7

Il est également possible de générer des familles 4/7 (il y a en a $7 \times 5 = 35$ au total), ce qui serait utile dans le cas où le seuil d'interférence σ est très élevé.

2.1.2 Statut des spots

Dans l'exemple précédent, nous avons distingué entre un spot allumé ou actif et un spot éteint. Or, tout spot comporte un certain nombre de zones géographiques (typiquement 2 ou 3 zones). Il existe donc plusieurs possibilités pour chaque spot allumé. Comme le niveau de puissance du signal utile dépend de la zone, ainsi que les niveaux d'interférences générées par les zones des autres spots, il se peut qu'une zone d'un spot ne vérifie pas le

[1]Sur une ligne donnée, il y a 7 choix possibles pour un spot éteint et, pour chaque option, il y a 3 choix possibles pour le deuxième spot éteint, selon qu'il y a 0, 1 ou 2 spots allumés entre 2 spots éteints

critère d'allocation (2.1), alors qu'une autre zone du même spot le vérifie. Nous introduisons la notion de "statut" d'un spot qui prend valeur dans l'intervalle $[0, 2^{nbZones(s)} - 1]$, où $nbZones(s)$ désigne le nombre de zones dans le spot s. Pour un spot contenant 3 zones, son statut peut prendre l'une des valeurs suivantes :

0 : le spot est éteint ;
1 : la zone 0 du spot vérifie le critère d'allocation et est donc allumable ;
2 : la zone 1 du spot vérifie le critère d'allocation et est donc allumable ;
3 : les zones 0 et 1 vérifient toutes les deux le critère d'allocation et sont donc allumables ;
4 : la zone 2 du spot vérifie le critère d'allocation et est donc allumable ;
5 : les zones 0 et 2 vérifient toutes les deux le critère d'allocation et sont donc allumables ;
6 : les zones 1 et 2 vérifient toutes les deux le critère d'allocation et sont donc allumables ;
7 : les zones 0, 1 et 2 vérifient toutes les trois le critère d'allocation et sont donc allumables.

L'introduction du statut d'un spot donne plus de liberté et de souplesse lors de l'allocation de ressources. Ainsi, si un spot ayant 3 zones a un statut 7, ceci veut dire que nous pouvons allouer les ressources à n'importe laquelle des 3 zones.

2.1.3 Simplification du calcul du critère d'allocation

Une autre conséquence de cette notion de spot est relative au calcul du critère d'allocation (2.1). Au lieu d'avoir à vérifier ce critère zone par zone, nous pouvons le faire au niveau des spots, ce qui diminue le nombre de fois où ce calcul doit être fait. Nous devons donc introduire les notions de gain et d'interférence au niveau des spots. Le gain sur un spot est égal au minimum des gains dans les zones allumables du spot, autrement dit $G(s) = \min_{z \text{ allumable dans le spot } s} G(z)$. L'interférence générée par un spot s' sur un spot s est $I(s, s') = \max_{z' \text{ allumable dans le spot } s'} I(s, z')$. Le niveau total d'interférence généré sur un spot s est $I(s) = \gamma S_1(s) + (1 - \gamma) S_2(s)$, où

$$\begin{aligned} S_1(s) &= \sum_{s' \text{ voisin et actif}} I(s, s'), \\ S_2(s) &= \sum_{s' \text{ actif}} I(s, s'). \end{aligned}$$

Ainsi un spot est *valide* si le critère suivant est vérifié :

$$\frac{G(s)}{I(s)} \geq \sigma, . \tag{2.2}$$

Considérons par exemple un spot ayant 3 zones et un statut 7. Ceci veut dire que les 3 zones du spot sont allumables. Pour vérifier la validité de cette hypothèse, il aurait fallu vérifier si les 3 zones vérifient le critère d'allocation donné en (2.1). Il est nettement plus avantageux de vérifier si le statut 7 est valide en utilisant le critère donné en (2.2) où $G(s)$ est la valeur minimale du gain sur le spot (le minimum entre les gains sur les zones).

Idéalement, nous voudrions avoir des spots allumés ayant le statut $2^{nbZones(s)}-1$ (toutes les zones d'un spot sont allumables). Pour ceci, nous allons, dans un premier temps, donner ce statut à tous les spots d'une famille. Pour tester la validité de la famille obtenue, il faudrait tester la validité de tous ses spots allumés, donc, utiliser le critère (2.2), avec $G(s) = G_{\min}(s) = \min_{z \text{ dans le spot } s} G(z)$, pour tous les spots.

2.1.4 Heuristique de génération de familles valides

Pratiquement, une famille est représentée par un vecteur, dont la taille est égale au nombre de spots, et dont les éléments sont les statuts des spots. Comme nous voulons maximiser le nombre de zones allumées, nous commençons par générer les 7 familles 6/7 dans lesquelles tout spot allumé s a le statut $2^{nbZones(s)}-1$, les spots éteints ayant le statut 0. Nous testons successivement la validité de ces familles et les séparons en deux tas : d'un coté les familles 6/7 valides et de l'autre coté les familles 6/7 non valides. Nous pouvons faire de même avec les familles 5/7, 4/7, etc., selon le seuil d'interférence σ considéré. Pour qu'une famille non valide devienne valide, il faudrait que certaines de ses zones allumées soient éteintes. Nous proposons l'heuristique suivante pour rendre une famille valide :

1. choisir au hasard une famille du tas de familles non valides ;
2. tant que la famille n'est pas valide :
   (a) choisir au hasard un spot,
   (b) si le statut est non nul et que le spot est non valide, éteindre au hasard une des zones allumées et sauvegarder le numéro du spot ainsi modifié ;
3. essayer pour un certain nombre de fois de rallumer des zones qui ont été éteintes en 2. et tester la validité de la famille après chaque essai : une modification n'est adoptée que si la famille est valide ;
4. comparer la famille valide obtenue en 3. à celles qui se trouvent dans le tas des familles valides. En cas de redondance, incrémenter un compteur de redondances et rejeter la famille, sinon, ajouter la famille au tas des familles valides. Revenir à 1. pour générer une autre famille.

Cet algorithme s'arrête soit quand le nombre voulu de familles valides a été atteint, soit quand le compteur de redondances atteint une certaine valeur maximale.

2.1.5 Notations

Pour l'application qui suit, une famille F_i sera ainsi caractérisée par la liste des statuts des spots utilisés lors d'une émission sur une fréquence donnée. Pour chaque spot, la famille précise quelles zones peuvent être utilisées par l'intermédiaire de la fonction F_i^A. On notera cependant que sur les zones utilisables d'un spot, une seule peut être utilisée en émission pour une fréquence donnée. L'utilisation de cette famille garantie que le niveau d'interférence globale est acceptable.

Le sens de F_i^A se déduit comme suit :

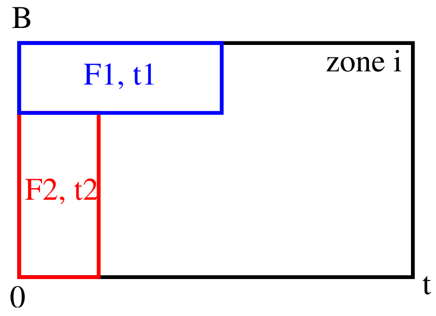

FIG. 2.3 – Utilisation de terminaux de types différents.

- $F_i^A(z) = on$: zone z allumable,
- $F_i^A(z) = off$: zone z éteinte.

2.2 Résolution des contraintes de placement des capacités sur le canal radio

Les contraintes au niveau du canal radio mettent en jeu deux grandeurs : la largeur de bande B et le temps. Lorsqu'on planifie l'utilisation d'un terminal de type t_k pendant un time-slot, on utilise schématiquement un rectangle d'une certaine aire dans l'intervalle temps-fréquence. La figure 2.3 illustre l'utilisation dans un même spot de deux terminaux de types différents.

Ainsi, si on identifie les types par des indices de 1 à τ ordonnés par largeur de bande décroissante, et si on note x^{t_i} le nombre de time-slots utilisés dans le spot en question par le type i, on a :

$$\sum_{i \in \{1 \dots \tau\}} x^{t_i} \leq \frac{BT}{\Delta}.$$

Le résultat suivant nous sert à établir les propriétés d'un remplissage de time-slots :

Lemme 1 *Soit $G = (V, E)$ le graphe orienté avec $V = \{1, \dots, \tau\}$ et $E = \{(j, k) : j < k\}$ avec $w_{(j,k)} = \frac{t_j^b}{t_k^b}$. Alors tout chemin dans G de 1 à τ a un poids inférieur à $w_{(1,\tau)}$.*

DÉMONSTRATION. On note que le graphe G est transitif. Par ailleurs, pour tous réels x et y tels que $x \geq 1$ et $y \geq 1$, on a $x - 1 + y - 1 = xy - 1 - (x-1)(y-1) \leq xy - 1$. Ainsi, si $(i, j) \in E$ et $(j, k) \in E$, alors $(i, k) \in E$ et $w_{(i,k)} \geq w_{(i,j)} + w_{(j,k)}$. D'où le résultat par transitivité. □

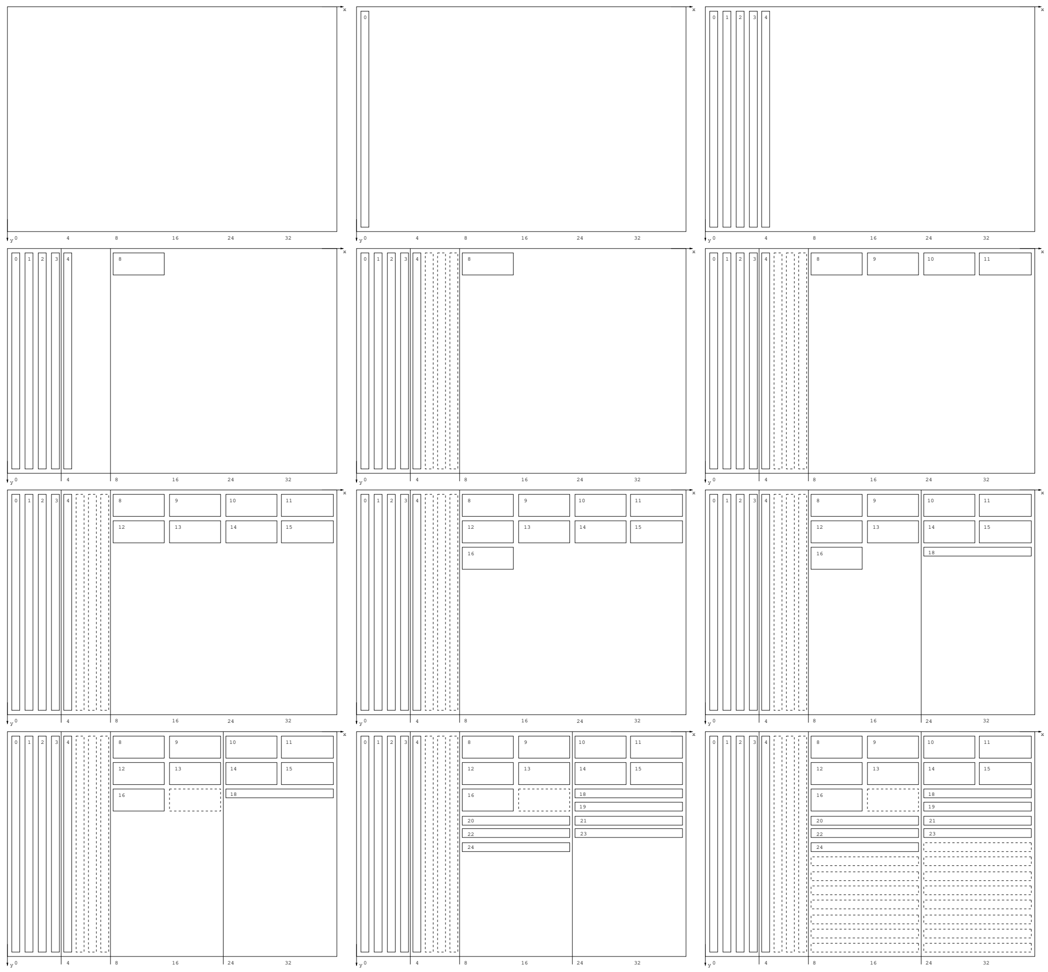

FIG. 2.4 – Etapes de l'algorithme de placement

Par la suite, on montre qu'un chemin dans ce graphe correspond en fait à des pertes à cause de la structure géométrique du problème.

Résultat 1 *Si* $\frac{B}{t_1^b}$ *est un entier, alors il est possible d'arranger dans la fenètre temps-fréquence les time-slots à hauteur de* x^{t_i}*, pour* $i \in 1, \ldots, \tau$ *dès que :*

$$\sum_{i \in \{1 \ldots \tau\}} x^{t_i} \leq \frac{BT}{\Delta} - \frac{t_1^b}{t_\tau^b} + 1.$$

DÉMONSTRATION. Nous donnons le squelette de la preuve. L'espace de la trame peut être découpé en "carrés$(1,\tau)$" donc le coté "fréquence" est la largeur de bande maximale et le coté "temps" est le temps de time-slot maximal. On remplit les carrés$(1,\tau)$ par ordre croissant de type.

Si un carré$(1,\tau)$ contient des types i et des types j on l'autorise à "perdre" l'espace $w_{(i,j)}$: en effet il se subdivise lui-même en carrés(i,j) qui vont être de types croissants de i à j.

Au total, au maximum on perd la somme des poids dans un chemin du graphe du lemme précédent. □

Par la suite on posera $\delta = \frac{t_1^b}{t_\tau^t} - 1$. On note que pour les données test du tableau 1.1, cette contrainte permet de résoudre le problème du placement en sacrifiant moins de 1% de la bande passante. Il est possible d'être encore plus performant en essayant le calcul ci-dessous avec une valeur de δ inférieure en faisant le pari raisonnable que l'arrangement devrait pouvoir se faire.

En pratique, on peut réaliser le placement suivant de nombreuses politiques différentes qui conduisent souvent à un gaspillage inférieur à celui de la preuve précédente. Ceci étant, l'écriture de l'algorithme n'est pas du tout trivial et reste fastidieux. Nous nous contentons de donner ici un exemple des différentes étapes d'un algorithme travaillant sur 3 types sur la figure 2.4. Les rectangles dessinés en pointillés sont des "espaces perdus" alors que les rectangles en traits continus sont des types placés sur le plan espace-temps. L'exemple choisi étant très petit (peu de fois le même nombre de type) et la configuration étant volontairement mauvaise, l'espace perdu est très important.

Les différentes étapes représentées sont basées sur un algorithme qui remplit de gauche à droite et qui "saute" vers l'ordre de multiplicité lorsqu'il y a un changement de type, comme précisé dans le squelette de la preuve. Les ordres de multiplicité de cet exemple sont 8 entre le type 1 et 2, puis 2 entre le type 2 et 3.

2.3 Réponse globale à la demande

La méthode proposée consiste, au lieu d'allouer des time-slots d'un certain type dans un slot, à allouer des familles typées simultanément dans l'ensemble des spots. Une famille est typée par *spot*. Si la famille F_i affecte le type t_j au spot s, on notera $F_i^T(z) = t_j$.

Considérons dans un premier temps les familles avec un seul type. Ainsi, pour une famille F_i on peut choisir un type de transmission t_k qui sera diffusé sur tous les spots concernés. Autrement dit, $\forall z \; F_i^T(z) = t_k$. On place cette transmission sur la largeur de bande au même endroit pour tous les spots concernés. De cette manière, le niveau d'interférence, à cet endroit de la bande est respecté. Sur une largeur de bande voisine, on peut utiliser une autre famille qui servira à satisfaire un autre (ou le même) type de demande.

Sur la figure 2.5, on montre un arrangement possible des ressources radio. Si $F_i^T(s) = t_k$, on note (F_i, t_k) dans le rectangle concerné, et cette structure se retrouve donc sur l'ensemble des zones allumées d'une famille (ici les zones des spots 0,1 et 3 pour la famille $F2$ et 0 et

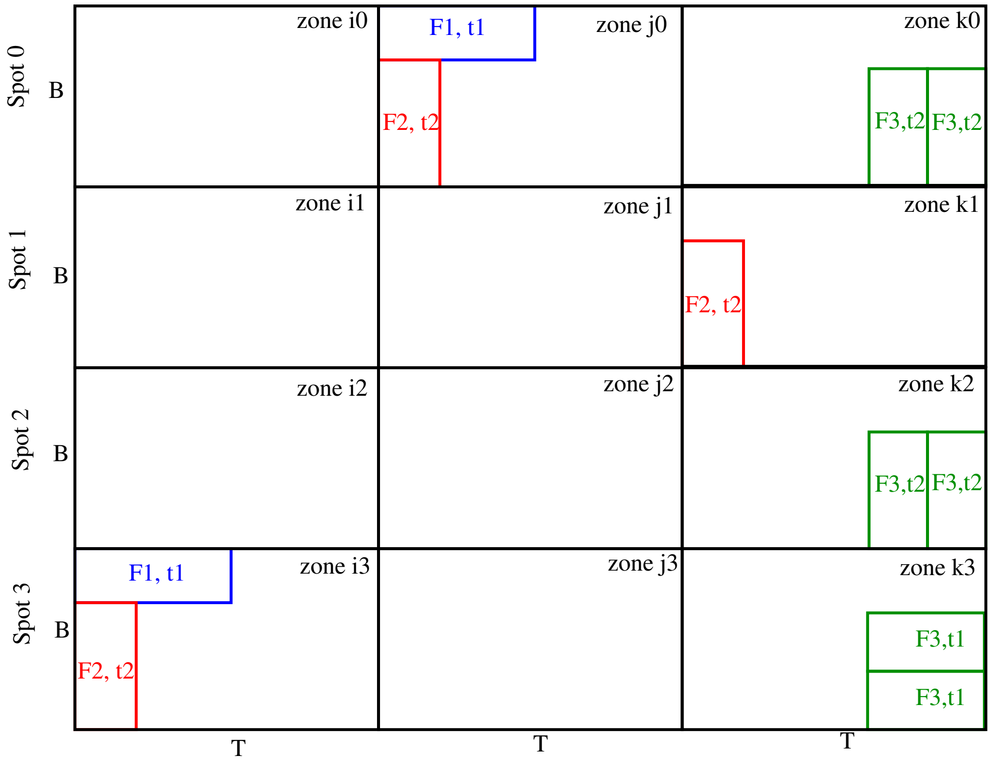

FIG. 2.5 – Un exemple global d'arrangement des familles.

3 pour la famille $F1$). On peut parler de bloc pour désigner le couple (F_i, t_k) occupant une certaine bande sur une zone donnée. Les contraintes de capacité sur chaque zone, en termes de bande et de temps (durée maximum d'une trame) est assurée par la contrainte d'aire d'un bloc (F_i, t_k) sur le pavé $B \times T$.

Une famille peut éventuellement recevoir plusieurs types suivant les spots. Dans ce cas elle correspond à des configurations réarrangeables du spectre. C'est le cas, par exemple pour les blocs (F_3, t_1) et (F_3, t_2). Ces familles ont un ordre de multiplicité spécifique. Si t_k est le type de la famille de plus large bande et $t_{k'}$ celui de largeur de bande la plus étroite, alors l'ordre de multiplicité de la famille est

$$F_i^M = \frac{t_k^b}{t_{k'}^b}.$$

2.4 Programme linéaire

$$\forall j,\ \forall z \in spot\ s \quad \sum_{i\ /\ \left| \begin{array}{rcl} F_i^T(s) & = & t_j \\ & et & \\ F_i^A(z) & = & on \end{array} \right.} F_i^M x_{F_i} \geq d(z,t_j) \tag{2.3}$$

$$\forall j,\ \forall z,z' \in spot\ s \quad \sum_{i\ /\ \left| \begin{array}{rcl} F_i^T(s) & = & t_j \\ & et & \\ & (& \\ F_i^A(z) & = & on \\ & ou & \\ F_i^A(z') & = & on \\ &) & \end{array} \right.} F_i^M x_{F_i} \geq d(z,t_j) + d(z',t_j) \tag{2.4}$$

$$\forall j,\ \forall s = (z,z',z''), \quad \sum_{i\ /\ \left| \begin{array}{rcl} F_i^T(s) & = & t_j \\ & et & \\ & (& \\ F_i^A(z) & = & on \\ & ou & \\ F_i^A(z') & = & on \\ & ou & \\ F_i^A(z'') & = & on \\ &) & \end{array} \right.} F_i^M x_{F_i} \geq d(z,t_j) + d(z',t_j) + d(z'',t_j) \tag{2.5}$$

$$\sum_i F_i^M x_{F_i} \leq \frac{T \times B}{\Delta} - \delta \tag{2.6}$$

- Contraintes de satisfaction de la demande : 2.3, 2.4, 2.5
- Contrainte d'aire : 2.6

Les variables utilisées sont donc les x_{F_i}, qui représentent le nombre de fois où sont utilisées les familles typées en question. Ce sont des variables entières.

Résultat 2 *Les équations 2.3, 2.4 et 2.5 garantissent la satisfaction des demandes en type t_j.*

Démonstration. La satisfaction des demandes en types t_j peut se calculer sur un flot depuis une source s, en passant par trois arcs (ou w, dans le cas où il y a w zones) de capacités respectives $d_1 = d(t_j, z_1)$, $d_2 = d(t_j, z_2)$, et $d_3 = d(t_j, z_3)$. Les capacités des autres arcs notées par $C[z_1, z_2, z_3]$, $C[z_i, z_j]$ pour $i \neq j$, $\{i,j\} \subset \{1,2,3\}$ et $C[z_i]$, $\{i\} \subset \{1,2,3\}$ sur la figure 2.6 sont données par :

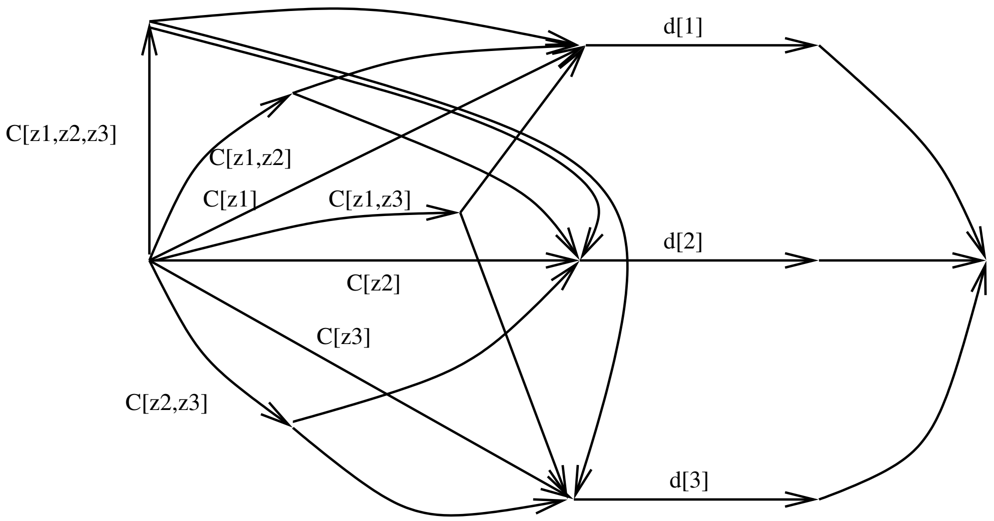

FIG. 2.6 – Modélisation des contraintes de zones par un flot.

$$\begin{aligned}
C[z_1, z_2, z_3] &= \sum_i x_{F_i}^{t_j} * U[Fi \textit{ peut allumer } z_1 \textit{ ou } z_2 \textit{ ou } z_3] \\
C[z_i, z_j] &= \sum_i x_{F_i}^{t_j} * U[Fi \textit{ peut allumer } z_i \textit{ ou } z_j \textit{ seulement}] \\
C[z_i] &= \sum_i x_{F_i}^{t_j} * U[Fi \textit{ peut allumer } z_i \textit{ seulement}]
\end{aligned}$$

où $U[Condition]$ vaut 1 quand la condition est vraie, 0 sinon.

Les capacités de tous les autres arcs de la figure sont supposées infinies. En effet, par le théorème de Ford Fulkerson [4] (ou dans sa version de Menger [6]), il existe un flot maximum et entier allant de la source au puits, qui est égal à la cardinalité d'une coupe minimale. Or les coupes de tailles finies sont au nombre de 8 (ou 2^w dans le cas de w zones), suivant le choix des arcs de capacité d_1, d_2 et d_3, et vérifier ces équations permet de vérifier que le flot est bien écoulé. L'une de ces équations est triviale puisqu'elle stipule que le flot des zones doit être inférieur à $d_1 + d_2 + d_3$. Les sept autres sont vérifiées par notre programme linéaire. □

2.4.1 Processus de génération des familles typées

Etant donné une famille non-typée F_i, il existe τ^N manières différentes de typer cette famille, soit bien plus qu'il n'est raisonnable de mettre dans notre programme linéaire ! Pour

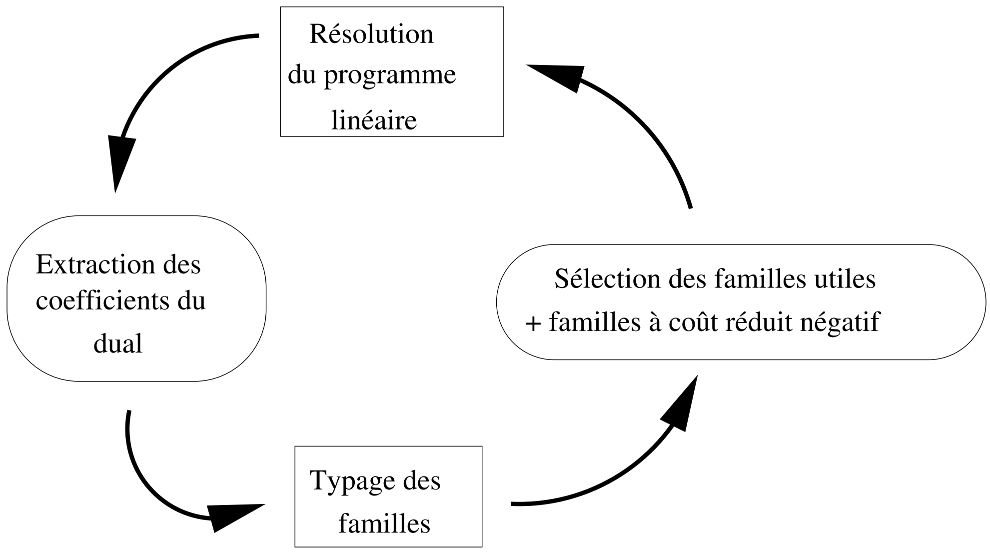

FIG. 2.7 – Processus de sélection des familles.

pouvoir s'en sortir on utilise un concept maintenant classique de *génération de colonnes* [2], illustré dans notre cas par la figure 2.7. Le programme commence avec des familles typées de manière homogène. Il est évident que si on supprime la contrainte d'aire une solution sera forcément obtenue. Dans le cas où cette solution ne satisfait pas la contrainte d'aire, l'aire occupée sera vue dans un premier temps comme une fonction objectif à minimiser. Dans l'autre cas, on cherchera par exemple à maximiser le revenu apporté par les ressources supplémentaires attribuées.

Etant donnée une certaine solution, un certain nombre de contraintes n du programme linéaire sont vérifiées à l'égalité. On s'intéresse alors à un système du type :

$$\begin{array}{ll} \text{Minimiser} & f = c \cdot x \\ \text{Tel que} & \begin{array}{ccc} Ax & = & b \\ x & \geq & 0 \end{array} \end{array}$$

On peut montrer qu'alors une solution existe avec seulement n variables x_{F_i} non-nulles[1, Theorem 9.3, page 145]. Désignons par A_B la matrice extraite du système d'équations correspondante, et x_B le vecteur des familles associées. Désignons par x_N le vecteur des autres familles, et A_N la matrice correspondante. De même on subdivise c en c_B et c_N. On a alors :

$$A_B x_B + A_N x_N = b \text{ et } f = c_B x_B + c_N x_N$$

et il vient :

$$\begin{array}{rcl} x_B & = & A_B^{-1}b - A_B^{-1}A_N x_N \\ f & = & c_B A_B^{-1}b + (c_N - c_B A_B^{-1}A_N)x_N \end{array}$$

La solution de base du système se trouve en posant $x_N = 0$ dans les équations ci-dessus. Le système est optimal si et seulement si

$$c_N - c_B A_B^{-1} A_N \geq 0.$$

Donc le système est améliorable si et seulement si on peut trouver un coefficient du vecteur ci-dessus négatif. Nous poussons la décomposition de A_N un peu plus en écrivant $A_N = [A_{\alpha_1} \dots A_{\alpha_j} \dots A_{\alpha_m}]$ où m est le nombre de colonnes de A_N, chacune correspondant à une famille d'indice α_j. On a en particulier $c_{\alpha_j} = F^M_{\alpha_j}$.

Résultat 3 *Pour toute famille non-typée $\mathcal{F}$, il existe une constante $K_{\mathcal{F}} > 0$ et une fonction $\kappa_{\mathcal{F}}$ de l'ensemble des couples $(type, spot)$ vers les réels positifs telles que pour toutes les familles typées issues de $\mathcal{F}$,*

$$c_i - c_B A_B^{-1} A_i = F_i^M \left(K_{\mathcal{F}} - \sum_{s \ spot} \kappa_{\mathcal{F}}(F_i^T(s), s) \right).$$

Démonstration. Notons que pour une certaine ligne de A correspondant à une famille F_i, notée A_i, tous les coefficients sont soit 0 soit F_i^M. Par ailleurs, si F_i et F_j sont deux typages d'une même famille non typée, alors $A_i/F_i^M = A_j/F_j^M$ et par ailleurs, si c_i et c_j sont les coefficients de c correspondant à F_i et F_j alors $c_i/F_i^M = c_j/F_j^M$ et on pose $K_{\mathcal{F}} := c_i/F_i^M$.

D'autre part on note que chaque spot correspond spécifiquement à certaines lignes de A, que l'on notera $P_s A$ (où P_s est la projection correspondante). Si F_i et F_j sont issues de la même famille non typée, et $F_i^T(s) = F_j^T(s)$, alors $P_s A_i / F_i^M = P_s A_j / F_j^M$, et on pose $\kappa_{\mathcal{F}}(F_i^T(s), s) := c_B A_B^{-1} P_s A_i / F_i^M$. D'où le résultat. □

On a alors le résultat suivant :

Résultat 4 *Le système ci-dessus est améliorable s'il existe une famille non-typée $\mathcal{F}$, avec*

$$K_{\mathcal{F}} - \sum_{s \ spot} \max_{t \ type} \kappa_{\mathcal{F}}(t, s) < 0.$$

En contrepartie, trouver une famille non-typée qui démontre que le système est améliorable permet d'améliorer la solution en introduisant la famille typée correspondante (avec le typage trouvé par la maximisation ci-dessus) dans le programme linéaire. Cette propriété réduit considérablement le nombre de recherches à faire pour atteindre l'optimal. En pratique, tant que l'on suppose que la solution est améliorable, on pourra aussi se restreindre à un choix d'un type parmi un intervalle $I \subseteq \{1, \dots, \tau\}$ afin de réduire le coefficient de multiplicité des familles obtenues et partant, la difficulté des contraintes d'intégrité.

2.5 Recherche de la solution flottante optimale

Dans le cas où le nombre de spots utilisé reste peu important, il est possible d'obtenir la solution optimale du problème flottant, en particulier en utilisant une recherche exhaustive des familles valides engendrant pour le problème ci-dessus une colonne à coût réduit négatif. Pour cela un prétraitement du dual est nécessaire. On sélectionne, pour chaque zone z le type qui lui donne un meilleur gain $g(z)$ au niveau du coût réduit. Il s'agit alors de trouver un ensemble de zones Z issues de spots distincts telles que

$$\sum_{z \in Z} g(z) > 1.$$

Chaque spot peut au plus être dans un état parmi quatre, s'il est éteint ou que l'une de ses trois zones est allumée. Ainsi cela nous donne, pour huit spots, un total de $4^8 = 65536$ configurations à tester, ce qui est très raisonnable. Ce nombre augmente rapidement avec le nombre de spots potentiellement actifs. Pour palier ce problème, et laisser une chance au programme de trouver un résultat optimal en temps rapide même en présence d'un nombre plus important de spots, un test élagué permettant de trouver les familles pouvant améliorer la solution plus facilement. Plusieurs étapes sont alors mises en œuvre, en particulier,

- Un "élagueur" sélectionne les zones au sein d'un spot. En particulier, si plusieurs zones se voient affecter le même gain, une parmi celles-ci est présélectionnée pour une recherche exhaustive ultérieure.
- Une fois les p familles de meilleur gain sélectionnées, celles-ci sont "réaugmentées" s'il est possible de rendre allumables plusieurs zones au sein d'un spot tout en respectant les contraintes de validité de la famille.

La figure 2.8 montre le processus de recherche élaguée.

2.6 Résolution pratique du programme linéaire

2.6.1 Découpage par nombre de types

Nous proposons de résoudre le programme linéaire $\mathcal{P}$, présenté en section 2.4, en utilisant la technique de génération de colonnes. La première étape consiste à résoudre la relaxation de ce problème $\mathcal{P}_{\mathcal{R}}$ pour déterminer rapidement une solution optimale flottante ainsi que les colonnes qui donnent cette solution. Nous résolvons chaque relaxation du problème en considérant les typages de famille n'utilisant qu'un nombre de type fixé. On a donc $\mathcal{P}_{\mathcal{R}}(1) \ldots \mathcal{P}_{\mathcal{R}}(k)$ problèmes à résoudre.

Plus précisement, $\mathcal{P}_{\mathcal{R}}(n)$ consiste à résoudre le programme linéaire de la section 2.4 avec

$$T \;/\; Card(F_i^T(z), \forall z) = n$$

Résoudre le problème relaxé en ne considérant que les familles avec un seul type revient donc à résoudre le problème $\mathcal{P}_{\mathcal{R}}(1)$.

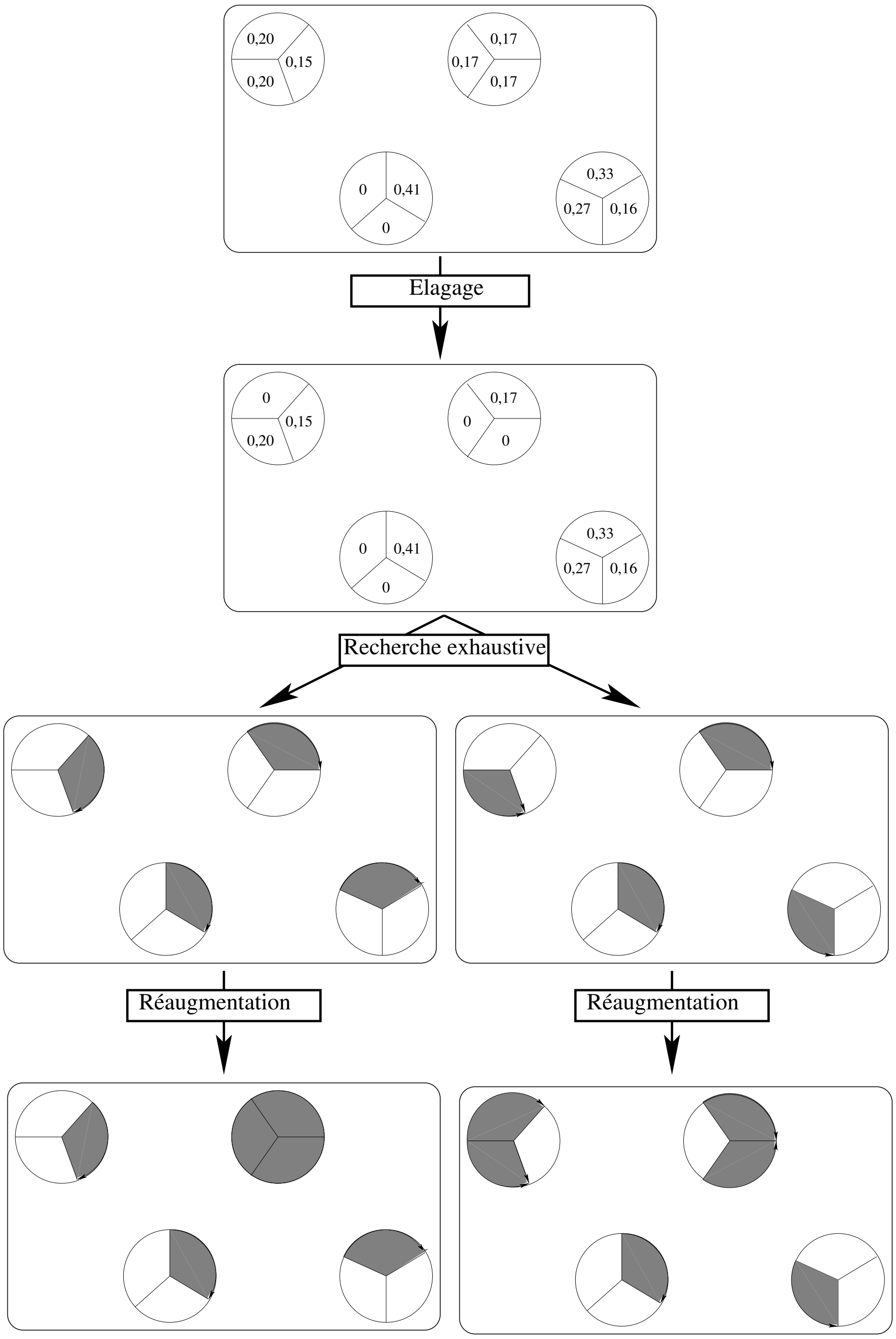

FIG. 2.8 – Recherche élaguée de familles améliorantes.

2.6.2 Résolution d'un problème relaxé $\mathcal{P}_{\mathcal{R}}(n)$

La première phase de résolution permet de résoudre le problème relaxé avec des familles à typage simple. Ainsi, une première solution admissible est trouvée pour $\mathcal{P}_{\mathcal{R}}(1)$. A partir de cette première solution, les meilleurs colonnes possibles sont générées à partir des coefficients du dual notés u. En effet, la meilleure colonne A_i que l'on puisse générer est donnée en résolvant le probème :

$$Trouver\ (c_i, A_i)\ qui\ minimize\ c_i - u^T A_i$$

Dans notre cas, c_i est connu et dépend du typage choisi : c'est la multiplicité du typage. Les coefficients de A_i qui sont admissibles sont soient nuls soient égaux à $c_i = F_i^M$. Ainsi, ce problème qui n'est en général pas trivial lors de la génération classique d'une colonne consiste dans notre cas à trouver les coefficients minimaux du dual, à chercher dans u^T.

Cependant, l'écriture des équations (2.3), (2.4), (2.5) conduit, lors de l'expression sous forme d'un programme linéaire de ces équations à écrire 7 équations pour un type j donné et un spot s donné. On obtient donc 7 coefficients du dual pour un couple (j, s) que nous sommons pour obtenir l'expression du gain pour un spot donné et un type donné noté : $u(j, s)$. La construction d'une colonne admissible A_i consiste donc à trouver les $u(j, s)$ minimaux pour chaque spot s parmi les $n < k$ types possibles.

$$\forall s \in S, A_i(j, s) = F_i^M\ tq\ j\ minimize\ \{u(j, s)\}$$

2.6.3 Résolution du problème $\mathcal{P}$

La résolution de $\mathcal{P}_{\mathcal{R}}(1)$ à $\mathcal{P}_{\mathcal{R}}(k)$ permet de récupérer les colonnes qui donnent la meilleure solution flottante dans chaque cas. Toutes ces colonnes sont alors introduites dans un nouveau programme linéaire en nombre entier et sont candidates pour donner la meilleure solution possible entière.

La résolution à l'aide du logiciel Cplex CONCERT permet de trouver une solution entière que nous fixons arbitrairement à 1% de la solution optimale de $\mathcal{P}_{\mathcal{R}}$. En pratique, résoudre complètement $\mathcal{P}$ avec les colonnes candidates ne termine pas en un temps raisonnable.

Chapitre 3

Résultats numériques

Nous présentons dans cette section des résultats obtenus avec la méthode décrite plus haut. Nous avons testé différents types de configurations entre 8 et 32 spots. La demande a été générée suivant les exemples précédemment fournis par ALCATEL SPACE INDUSTRIES, et les Interférences ainsi que les gains, ont été tirés suivant une loi uniforme en dB, selon les spécifications données par le document ALCATEL SPACE INDUSTRIES. L'interférence a été choisie, pour 85% de sa valeur, comme une interférence de voisinage (les six spots voisins) et pour 15%, comme une interférence globale ($\gamma = 0.85$).

La sortie "classique" de nos programmes est un plan d'"allocation temps/fréquence, comme celui indiqué sur la figure 3.1.

3.1 Résultats avec 8 spots

Dans ce cas, avec des temps de calcul de l'ordre de la minute sur une machine du type Pentium III, nos programmes peuvent trouver la solution flottante optimale. Ce résultat est particulièrement satisfaisant, car il permet d'analyser avec précision les causes d'interférence au niveau des spots. Sur la figure 3.2 nous montrons les valeurs d'aires minimales trouvées en fonction du seuil. La figure met clairement en évidence le fait qu'à certaines valeurs pivot, la valeur d'aire consommée augmente brutalement. Cela est dû au fait qu'à partir de certaines valeurs de seuil, des familles ne sont plus valides et leur absence dégénère la solution. Le tableau 3.1 détaille leurs structures.

Ce résultat a évidemment un impact très fort sur la conception des antennes, puisque l'on est capable de mettre en évidence les configurations d'interférences qui entravent la génération de bonnes solutions.

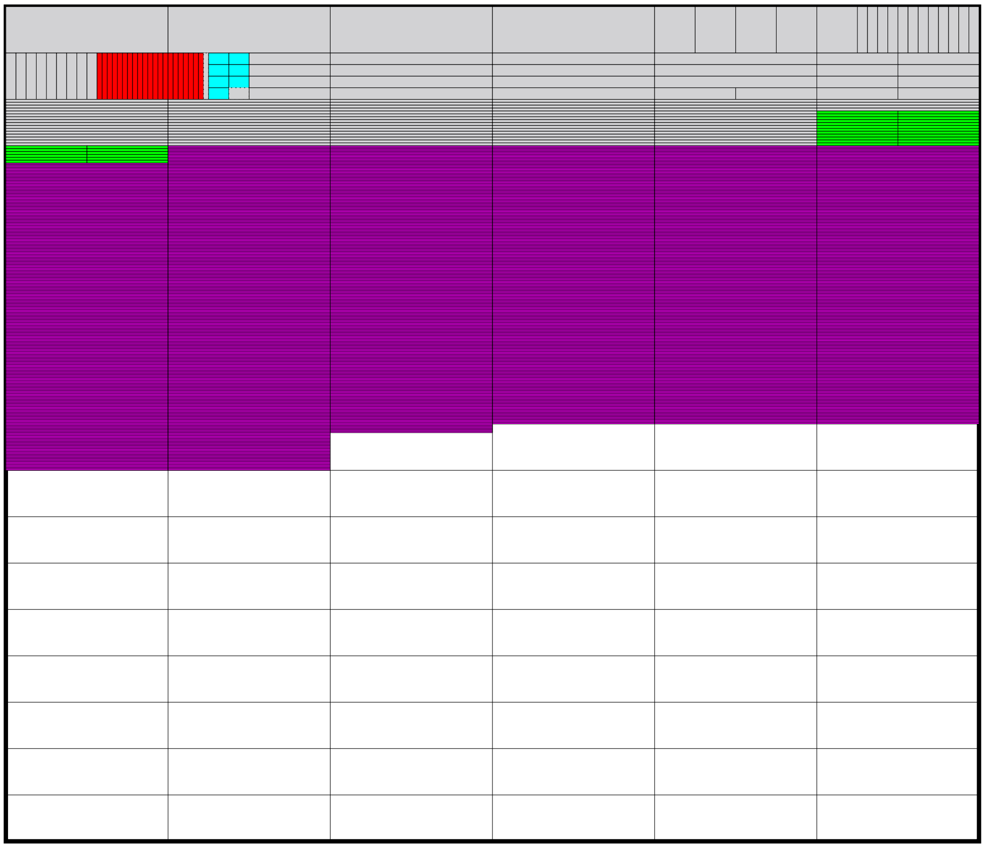

FIG. 3.1 – Un exemple de résultat d'allocation de ressources.

Seuil	Spot 0			Spot 1			Spot 2			Spot 3			Spot 4			Spot 5			Spot 6			Spot 7		
	0	1	2	0	1	2	0	1	2	0	1	2	0	1	2	0	1	2	0	1	2	0	1	2
373	X					X						X				X								
373	X					X						X						X						
418	X					X						X												
423	X					X										X								
450		X		X							X					X								
450		X		X								X					X							
450		X		X							X										X			
469			X	X								X					X							
472			X	X							X					X								
472			X	X							X										X			
472				X							X					X					X			
472					X						X					X					X			
490		X		X							X													
496				X								X				X					X			
501			X	X								X				X								
505			X	X												X					X			
510				X				X										X						

TAB. 3.1 – Familles perdues en fonction des seuils.

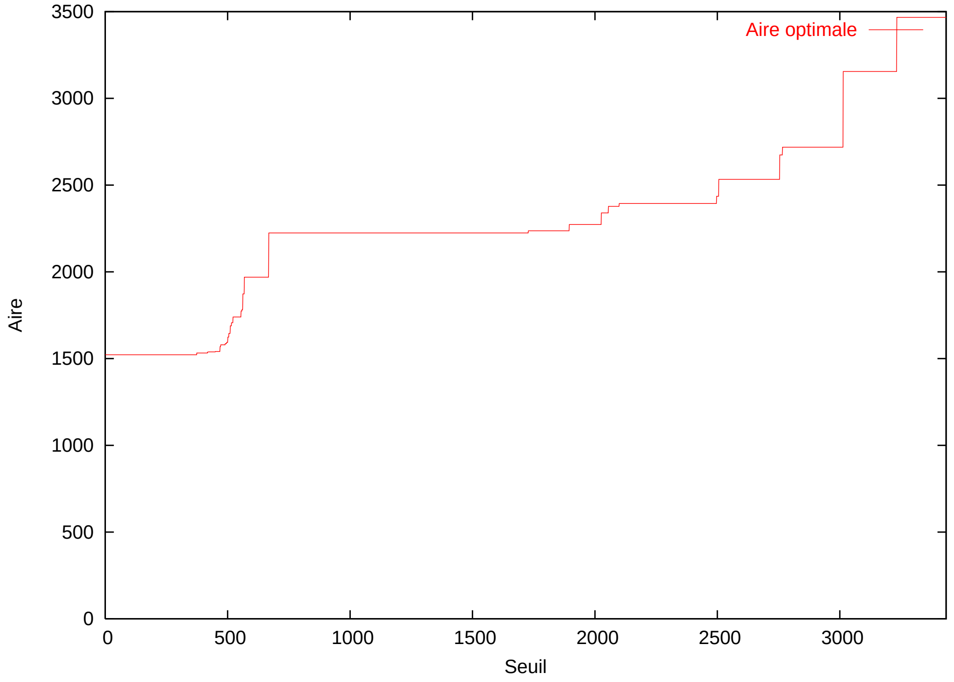

FIG. 3.2 – Valeurs d'aire obtenues sur l'ensemble des seuils possibles.

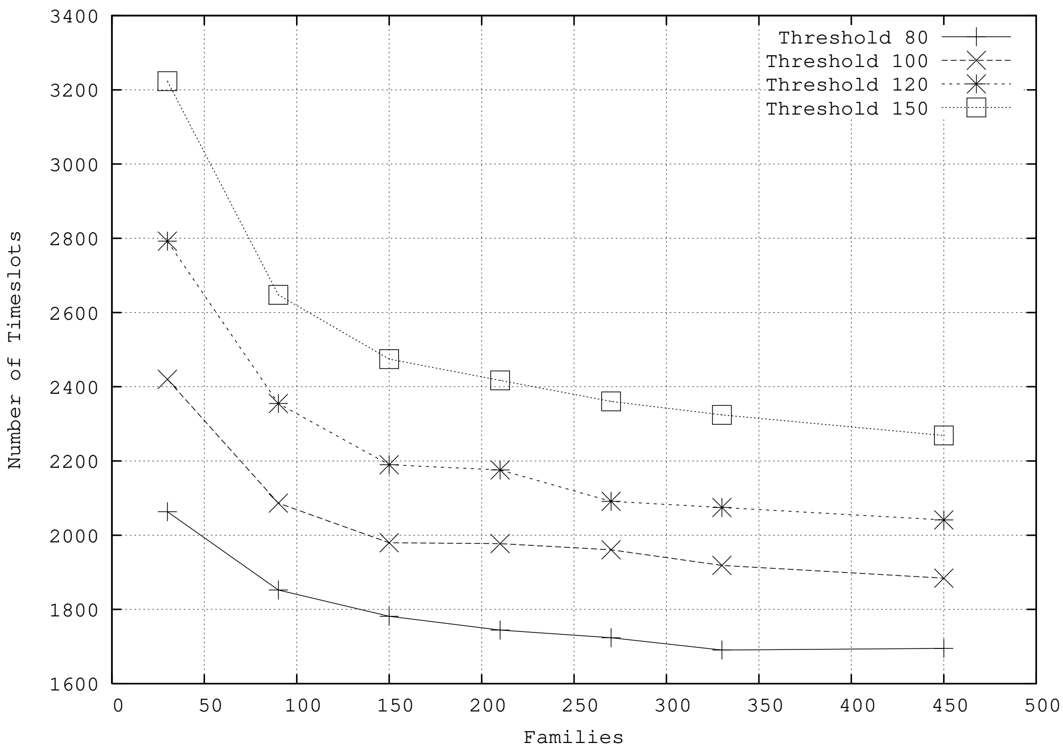

FIG. 3.3 – Aires obtenues sur un exemple pour différents nombres de familles aléatoires.

3.2 Résultats avec 32 spots

Pour une configuration de 32 spots, nous préconisons un fonctionnement non-optimal partant d'un nombre de familles limité. Des tests ont montré que la procédure de recherche d'une solution flottante optimale pouvait grandement améliorer la solution. Néanmoins, le nombre de configurations possibles rendent alors les résultats peu présentables. C'est pourquoi nous avons décidé de simplement mettre en évidence des résultats avec des familles valides générées aléatoirement. Leur nombre tend à augmenter le temps de résolution tout en améliorant la solution. Il est donc possible de trouver un compromis pour l'élaboration d'une solution en temps réel. Les résultats sont montrés sur les figures 3.3 et 3.4.

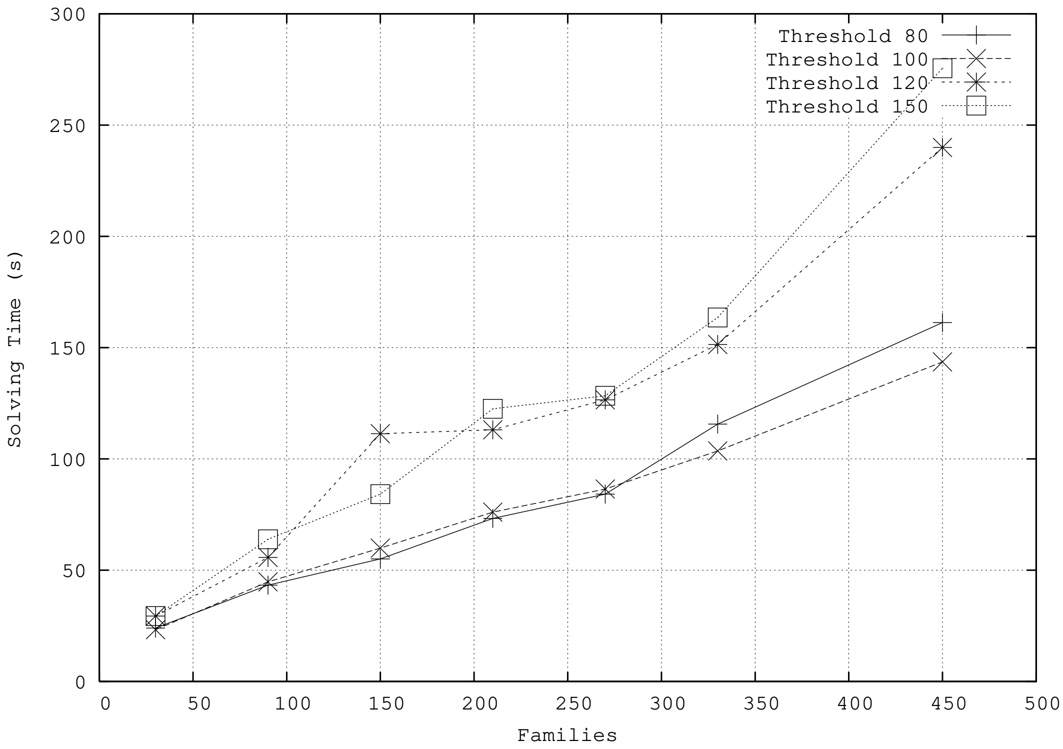

FIG. 3.4 – Temps de calcul obtenus sur un exemple pour différents nombres de familles aléatoires.

Bibliographie

[1] V. Chvatal. *Linear programming.* W. H. Freemann and Company, 1983.

[2] G. B. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations Research*, 8 :101–111, 1960.

[3] T. ElBatt and A. Ephremides. Frequency reuse impact on the optimum channel partitioning for hybrid wireless systems. *International Mobile Satellite Conference (IMSC)*, 1999.

[4] L.R. Ford and D.R. Fulkerson. *Flows in Networks.* Princeton University Press, Princeton, NJ, 1962.

[5] M. Grötschel, L. Lovász, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1 :169–197, 1981.

[6] K. Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, pages 96–115, 1927.

Remerciements

Nous remercions particulièrement Benoit Fabre, Cécile Guiraud et Isabelle Buret, d'Alcatel Space industries, pour leurs nombreuses explications techniques, leur contribution à la modélisation du problème, ainsi que l'intéret qu'ils ont porté à sa résolution.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399