



HAL
open science

Comment appliquer les chaînes augmentantes pour atterrir à l'heure ?

Nicolas Nisse, Alexandre Salch, Valentin Weber

► **To cite this version:**

Nicolas Nisse, Alexandre Salch, Valentin Weber. Comment appliquer les chaînes augmentantes pour atterrir à l'heure ?. ALGOTEL 2015 - 17èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, Jun 2015, Beaune, France. hal-01144674

HAL Id: hal-01144674

<https://hal.science/hal-01144674>

Submitted on 22 Apr 2015

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

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

Comment appliquer les chaînes augmentantes pour atterrir à l'heure ?[†]

Nicolas Nisse¹ and Alexandre Salch² and Valentin Weber²

¹Inria & Univ. Nice Sophia Antipolis, CNRS, I3S, UMR 7271, 06900 Sophia Antipolis, France

²Innovation & Research, Amadeus IT Group SA

Lorsqu'un avion approche d'un aéroport, il dispose d'un intervalle de temps (*slot*) très limité (une vingtaine de minutes) pour atterrir. Si l'avion a du retard (à cause des conditions météorologiques, à cause du retard d'autres avions, ou si lui-même a eu du retard au décollage), il perd son slot et il faut qu'un nouveau slot lui soit attribué par les contrôleurs des opérations de la compagnie aérienne. Cependant, les slots d'atterrissage sont une denrée rare et, pour qu'un avion A puisse atterrir à l'heure, les contrôleurs doivent régulièrement modifier l'attribution des slots d'autres avions afin d'affecter un slot compatible avec l'heure d'arrivée de l'avion A . Ce problème peut aisément être modélisé comme un problème de couplage dans un graphe biparti. Malheureusement, dû au système mis en place pour permettre ces échanges, les contrôleurs aériens ne peuvent effectuer leurs modifications qu'en effectuant deux types d'opérations : soit attribuer à l'avion A un slot libre, soit donner à l'avion A le slot d'un avion B et attribuer un slot libre à ce dernier.

Le problème devient donc le suivant. Soit G un graphe et M un couplage (ensemble d'arêtes deux-à-deux disjointes) de G . Comment calculer un couplage maximum pouvant être obtenu à partir de M en utilisant uniquement des chemins augmentants de longueur au plus k ? Ce problème a déjà été étudié dans le cadre des réseaux sans-fil car il fournit une simple approximation au problème de couplage maximum. Nous prouvons que, pour $k = 3$, ce problème peut être résolu en temps polynomial, fournissant ainsi un algorithme efficace pour les contrôleurs aériens. Nous prouvons ensuite que, pour tout entier impair $k \geq 5$, le problème est NP-complet dans les graphes bipartis planaires de degré au plus 3.

Keywords: couplage dans les graphes, chemins augmentants

1 Introduction

Pour des raisons évidentes de sécurité, les autorités aéroportuaires doivent s'assurer que le nombre d'avions atterrissant dans une plage horaire donnée soit inférieur à la capacité de l'aéroport. Ainsi, les aéroports disposent d'intervalles de temps (les *slots*) qui sont initialement attribués aux avions en fonction de leur planning d'arrivée. Bien sûr, un avion ne peut se voir affecter qu'un slot qui soit compatible avec son heure d'arrivée. Cependant, si un avion A prend du retard (« pour une raison indépendante de notre volonté »), les contrôleurs des opérations doivent lui attribuer un nouveau slot via le système d'information gérant ces échanges. Les réglementations strictes que respecte ce système font que seules deux opérations sont possibles [MB10]. Soit un slot disponible est attribué à l'avion A , soit un slot S déjà assigné à un autre avion B est réattribué à A alors que B se voit attribuer un slot disponible S' . Dans les deux cas, le slot S doit être compatible avec les horaires de A et dans le second cas, S et S' doivent être compatibles avec les horaires de B . Un exemple de scénario simple est représenté sur la Figure 1. Si plusieurs avions prennent du retard et perdent leurs slots, la résolution de ces problèmes devient difficile pour les contrôleurs des opérations qui ne disposent pas d'outils pour réaliser ces modifications et doivent s'assurer « à la main » que tous les avions vont récupérer un slot.

Le problème de *réaffectation des slots* peut évidemment être modélisé par un problème de couplage dans les graphes. Soit un graphe $G = (V, E)$. Un *couplage* M de G est un ensemble d'arêtes deux-à-deux disjointes. Un sommet v est dit *couvert* par M si il existe $e \in M$ tel que $v \in e$. Dans le cas contraire, v est dit *exposé*. Le problème du calcul de couplage maximum a été très étudié et il est bien connu qu'il peut être résolu en temps polynomial [Edm65]. On note $\mu(G)$ la taille d'un couplage maximum de G . Un ingrédient clé dans

[†]Le version complète de l'article ainsi que les preuves peuvent être trouvées ici [NSW15]

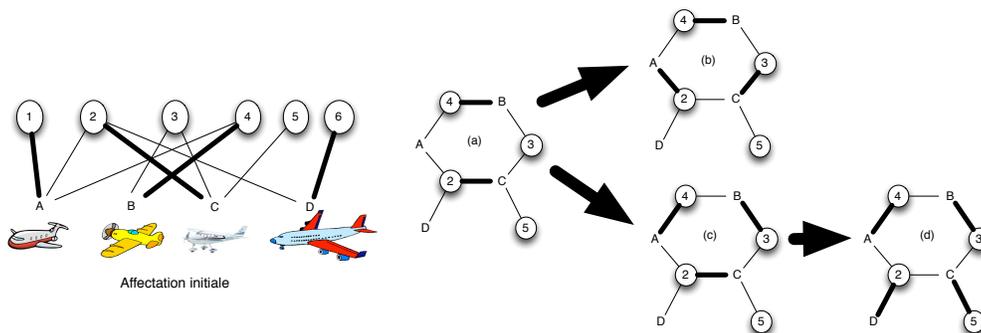


FIGURE 1: Exemple de scénario avec 6 slots et 4 avions. Chaque avion n'est relié qu'aux slots qui lui sont compatibles. Initialement (gauche), les slots 1, 2, 4 et 6 sont attribués aux avions A, C, B et D respectivement (les arêtes du couplage sont représentées en gras). Après un retard des avions A et D , les slots 1 et 6 ne leur sont plus accessibles (a). Si à partir de (a), le chemin $(A2C3)$ est augmenté (le slot 2 est réassigné à A et 3 est assigné à C), on obtient la configuration (b) où aucune réaffectation n'est possible (il n'existe aucun chemin augmentant de longueur au plus 3). Une autre solution est d'augmenter le chemin $(A4B3)$ (pour atteindre (c)), puis $(D2C5)$ (Fig. (d)) où tous les avions peuvent atterrir.

la plupart des travaux sur les couplages est la notion de *chemin augmentant*. Un chemin $P = (v_0, \dots, v_k)$ est dit M -augmentant si, v_0 et v_k sont exposés et, pour tout $0 \leq i < k$, $e_i = \{v_i, v_{i+1}\} \in M$ si et seulement si i est impair. En particulier, un chemin augmentant a toujours une longueur k impaire. Un théorème classique dû à Claude Berge dit qu'un couplage M est maximum si et seulement si il n'existe pas de chemins M -augmentants. En particulier, il est facile de vérifier que si P est un chemin M -augmentant, alors $M \Delta E(P)$ est un couplage de taille $|M| + 1$, avec $E(P)$ l'ensemble des arêtes de P et Δ la différence symétrique. Lorsque l'on passe du couplage M à $M \Delta E(P)$, on dit que l'on *augmente* le chemin M -augmentant P .

Le problème de réaffectations des slots décrit dans le premier paragraphe peut être modélisé ainsi. Soit un graphe biparti $G = (X \cup Y, E)$. La partie X représente l'ensemble des avions et Y est l'ensemble des slots disponibles. Il existe une arête entre $a \in X$ et $s \in Y$ si le slot s est compatible avec l'horaire d'arrivée de l'avion a . Soit M un couplage de G qui représente une assignation initiale de slots à certains avions (voir Fig 1 (a)). Le problème de réaffectation des slots est alors équivalent à celui de calculer un couplage maximum qui peut être obtenu à partir de M en augmentant des chemins de longueur au plus 3.

Contributions. Nous considérons le problème dont les entrées sont un graphe $G = (V, E)$, un couplage $M \subseteq E$ et un entier impair $k \geq 1$. Soit $\mu_k(G, M)$ la taille maximum d'un couplage de G qui peut être obtenu à partir de M en augmentant uniquement des chemins augmentants de longueur au plus k . Le problème consiste à calculer une séquence de chemins augmentants de longueur au plus k qui permettent d'obtenir, à partir de M , un couplage de taille $\mu_k(G, M)$. Insistons sur le fait que seuls des chemins augmentants de longueur impaire peuvent être augmentés. Dans les cas $k \in \{1, 3\}$, nous prouvons que la complexité du problème est équivalente à celle du couplage maximum classique (sans restriction de longueur) et donc qu'il peut être résolu en temps polynomial. Nous proposons un algorithme polynomial qui calcule une séquence de chemins augmentants de longueur au plus 3 pour obtenir un couplage maximum dans ce contexte, aidant ainsi nos amis contrôleurs. Dans un second temps, nous prouvons que, pour tout $k \geq 5$ impair, le problème est NP-complet dans les graphes bipartis planaires de degré maximum au plus 3.

État de l'art. Le premier algorithme polynomial pour résoudre le problème du couplage maximum dans les graphes généraux est dû à Edmonds [Edm65]. Par la suite de nombreux travaux ont proposé des algorithmes plus efficaces [HK73, MV80, DP14]. En particulier, les algorithmes dans [HK73, MV80] sont basés sur l'augmentation de chemins augmentants dans l'ordre croissant de leurs longueurs. Une telle méthode donne une bonne approximation puisque n'augmenter que les chemins de longueur au plus $2k - 3$ fournit une $(1 - 1/k)$ -approximation du problème de couplage maximum [HK73].

Le problème de couplage avec des chemins augmentants de longueur bornée a été étudié dans le contexte des réseaux sans-fil car il fournit une bonne approximation du couplage maximum à l'aide d'un algorithme glouton simple. En particulier, il permet d'obtenir des algorithmes distribués pour le calcul de l'ordonnement des transmissions sujettes à des interférences [WS05, BSS09].

Comment appliquer les chaînes augmentantes pour atterrir à l'heure ?

2 Couplage maximum et chemins augmentants de longueur bornée

Il est facile de vérifier que, dans les cas $M = \emptyset$ ou $k = 1$ (M quelconque), le problème de réassignation peut être résolu de façon équivalente au problème de couplage maximum classique. Lorsque $k \geq 3$, la difficulté provient du fait que l'ordre dans lequel les chemins sont augmentés est important. Cela est illustré sur la Figure 1 où augmenter d'abord le chemin (A2C3) mène à une configuration non optimale qui ne peut plus être améliorée. En particulier, l'ordre dans lequel les chemins sont augmentés influe sur la création ou non de nouveaux chemins qui peuvent être augmentés. Par exemple, pour $k = 5$, soit le graphe composé d'un chemin $(1, \dots, 7)$ et de 3 arêtes $\{5, 8\}$, $\{8, 9\}$ et $\{9, 10\}$. Le couplage initial est $\{\{2, 3\}, \{4, 5\}, \{8, 9\}\}$. Initialement, il y a 2 chemins augmentants de longueur au plus 5 : $P_1 = (6, 7)$ et $P_2 = (1, \dots, 6)$. Si l'on augmente P_1 , alors il ne reste aucun chemin augmentant de longueur au plus 5. Au contraire, augmenter P_2 « détruit » le chemin P_1 , mais « crée » le chemin $P_3 = (10, 9, 8, 5, 6, 7)$ qui peut être augmenté.

Théorème 1 Soient un graphe G et un couplage M , une séquence de chemins augmentants de longueur au plus 3 pour obtenir, à partir de M , un couplage de taille $\mu_3(G, M)$ peut être calculée en temps polynomial.

Idée de preuve. La différence entre les cas $k \leq 3$ et ≥ 5 est que dans le premier cas, il est possible d'ignorer les chemins augmentants créés au cours de l'augmentation de M . Soit $\mathcal{P}_k(G, M)$ l'ensemble des chemins M -augmentant de longueur au plus k dans le graphe G avec le couplage initial M . Pour $k = 3$, nous prouvons (Propriété $(*)$) qu'il est toujours possible d'obtenir un couplage de taille $\mu_3(G, M)$, à partir de M , en augmentant uniquement des chemins M -augmentants dans $\mathcal{P}_3(G, M)$, c'est-à-dire sans utiliser les nouveaux chemins augmentants qui pourraient être créés lors de l'augmentation d'autres chemins. Au contraire, dans l'exemple précédent pour $k = 5$, il est impératif de créer le chemin augmentant P_3 qui n'appartient pas à $\mathcal{P}_3(G, M)$ et de l'augmenter afin d'obtenir un couplage optimal. Cette propriété $(*)$ nous permet de prouver que $\mu_3(G, M)$ peut être facilement calculé à partir de $\mu_3(H, M')$ où H est un graphe et M' est un couplage de H qui peuvent être calculés en temps polynomial et $\mu_3(H, M')$ peut être calculé comme un couplage maximum « classique ».

Plus précisément, étant donné G et le couplage M , toute arête $e \notin M$ qui est incidente à 2 arêtes de M (arête de type 1) et toute arête $e \in M$ qui est incidente à un sommet de degré 1 dans G (arête de type 2) ne peuvent être dans un chemin de $\mathcal{P}_3(G, M)$. Soit (H, M') les graphe et couplage obtenus de (G, M) en supprimant les arêtes de type 1 (en conservant les sommets) et en supprimant récursivement les arêtes de type 2 (arêtes et sommets). Soit x le nombre d'arêtes de type 2 supprimées pour obtenir H . Grâce à la Propriété $(*)$, nous prouvons que $\mu_3(G) = \mu_3(H) + x$. Finalement, pour tout graphe G' et couplage M' dont toutes les arêtes ne sont ni de type 1 ni de type 2 (comme c'est le cas pour H), nous prouvons que $\mu_3(G', M')$ peut être calculé par un algorithme de couplage maximum « classique ». \square

Théorème 2 Pour tout $k \geq 5$ impair, calculer $\mu_k(G, M)$ est NP-complet dans la classe des graphes G bipartis planaires et de degré maximum au plus 3.

Idée de preuve. La preuve consiste en une réduction de 3-SAT planaire. Soit Φ une instance de 3-SAT planaire avec $n \in \mathbb{N}$ variables v_1, \dots, v_n et $m \in \mathbb{N}$ clauses C_1, \dots, C_m . Pour tout $i \leq n$, soit o_i le nombre de clauses dans lesquelles la variable v_i apparaît (positivement ou négativement). Nous construisons un graphe G (biparti planaire et de degré maximum 3) et un couplage initial M de G tel que $\mu_k(G, M) = |M| + m + \sum_{i \leq n} o_i$ si et seulement si Φ peut être satisfaite. En bref, pour $k \geq 5$, le graphe G est défini ainsi.

Pour toute variable v_i qui apparaît dans la clause C_j , $i \leq n$, $j \leq m$, soit G_{ij} le graphe obtenu d'un sommet c_{ij} , de deux chemins (t_{ij}, \dots, p_{ij}) et (n_{ij}, \dots, f_{ij}) de longueur $k - 1$ et d'un chemin (w_{ij}, \dots, v_{ij}) de longueur $k - 2$ en ajoutant les arêtes $\{c_{ij}, p_{ij}\}$, $\{c_{ij}, n_{ij}\}$ et une arête $\{p_{ij}, w_{ij}\}$ (resp., une arête $\{n_{ij}, w_{ij}\}$) si v_i apparaît positivement (resp., négativement) dans C_j . Le graphe G_{ij} est représenté sur la Figure 2 (gauche) avec les arêtes du couplage initial représentées en gras. Ce sous-graphe contient seulement deux chemins augmentants de longueur au plus k : T_{ij} le chemin de t_{ij} à c_{ij} et F_{ij} le chemin de c_{ij} à f_{ij} . Intuitivement, augmenter T_{ij} (resp., F_{ij}) revient à assigner la Variable v_i à Vrai (resp., à Faux). Dans un second temps, les graphes G_{ij} correspondant à la variable v_i sont combinés en un « cycle ». Soit $J_i = \{j_1^i, \dots, j_{o_i}^i\}$ l'ensemble des indices $j \leq m$ tels que v_i apparaît dans C_j . A partir d'une copie de G_{ij} pour chaque $j \in J_i$, les sommets $t_{ij_1^i}$ et $f_{ij_1^i}$ sont identifiés et, pour tout $1 \leq \ell < o_i$, les sommets $f_{ij_\ell^i}$ et $t_{ij_{\ell+1}^i}$ sont identifiés. Cela permet de

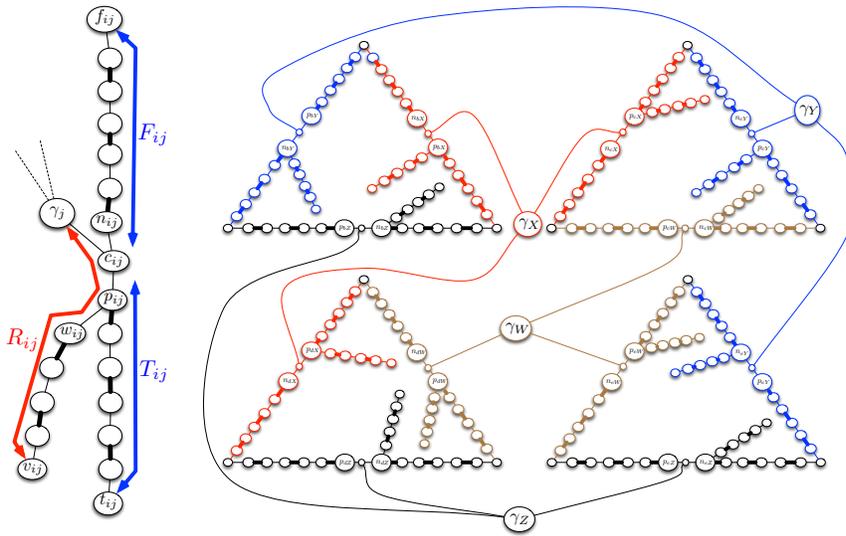


FIGURE 2: $k = 7$. (gauche) Graphe G_{ij} et Clause-sommet γ_j pour la variable v_i apparaissant positivement dans la Clause C_j . (droite) Exemple de réduction pour $\Phi = (b \vee c \vee d) \wedge (\bar{b} \vee c \vee \bar{e}) \wedge (\bar{c} \vee d \vee e) \wedge (\bar{b} \vee \bar{d} \vee \bar{e}) = X \wedge Y \wedge W \wedge Z$ avec Variables b, c, d, e et Clauses $X = b \vee c \vee d$ (rouge), $Y = \bar{b} \vee c \vee \bar{e}$ (bleu), $W = \bar{c} \vee d \vee e$ (marron) et $Z = \bar{b} \vee \bar{d} \vee \bar{e}$ (noir). Les arêtes du couplage initial sont représentées en gras.

s'assurer que, pour obtenir un couplage optimal, soit tous les chemins T_{ij} , $j \in J_i$, sont augmentés, soit tous les chemins F_{ij} , $j \in J_i$, sont augmentés. En d'autres termes, un couplage optimal correspond à une assignation des variables. L'ordre de J_i dépend du plongement planaire de Φ pour s'assurer de la planarité du graphe. Ajoutons un sommet γ_j par Clause C_j , $j \leq m$ et, pour tout $i \leq n$ tel que la variable v_i apparaît dans C_j , une arête $\{\gamma_j, c_{ij}\}$. Le point clé est que, si v_i apparaît positivement (resp., négativement) dans C_j , alors augmenter T_{ij} (resp., F_{ij}) crée un nouveau chemin augmentant R_{ij} de γ_j à v_{ij} (voir ex. sur Fig. 2 (droite)).

Si Φ est satisfiable, il est possible d'obtenir un couplage de taille $|M| + m + \sum_{i \leq n} o_i$ à partir du couplage initial M en augmentant, pour tout v_i assignée à Vrai (resp., à Faux), les chemins T_{ij} (resp., F_{ij}) pour tout $j \in J_i$. Puis, pour tout $j \leq m$, soit v_i une variable satisfaisant la Clause C_j , le chemin R_{ij} peut être augmenté. Réciproquement, nous prouvons que la seule façon d'obtenir un couplage de taille $|M| + m + \sum_{i \leq n} o_i$ est celle décrite ci-dessus, correspondant à une assignation des variables qui satisfait Φ . \square

Travaux futurs. La complexité du problème dans d'autres classes de graphes (arbres, graphes cordaux...) reste ouverte. Par ailleurs, est-ce que le problème devient plus facile si l'on s'autorise, en plus de l'augmentation de chemins de longueur bornée, à inverser le couplage de cycles alternants de longueur bornée ?

Références

- [BSS09] L. Bui, S. Sanghavi, and R. Srikant. Distributed link scheduling with constant overhead. *IEEE/ACM Trans. Netw.*, 17(5) :1467–1480, 2009.
- [DP14] R. Duan and S. Pettie. Linear-time approximation for maximum weight matching. *J. ACM*, 61(1), 2014.
- [Edm65] J. Edmonds. Paths, trees, and flowers. *Canadian Journal of Mathematics*, 17 :449–467, 1965.
- [HK73] J. E. Hopcroft and R. M. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM J. Comput.*, 2(4) :225–231, 1973.
- [MB10] D. Michalek and H. Balakrishnan. Dynamic reconfiguration of terminal airspace during convective weather. In *49th Conference on Decision and Control*, pages 4875–4881. IEEE, 2010.
- [MV80] S. Micali and V. V. Vazirani. An $O(\sqrt{|V||E|})$ algorithm for finding maximum matching in general graphs. In *21st Symp. on Foundations of Comp. Sc. (FOCS)*, pages 17–27. IEEE, 1980.
- [NSW15] N. Nisse, A. Salch, and V. Weber. Recovery of disrupted airline operations, 2015. <http://www-sop.inria.fr/members/Nicolas.Nisse/publications/RR-8679.pdf>.
- [WS05] X. Wu and R. Srikant. Regulated maximal matching : A distributed scheduling algorithm for multi-hop wireless networks with nodeexclusive spectrum sharing. In *IEEE Conf. on Decision and Control*, 2005.