RenPar'17 / SympA'2006 / CFSE'5 / JC'2006
Canet en Roussillon, 4 au 6 octobre 2006

Satisfaction de requêtes par affectation de dates d'émissions dans les réseaux radios

Benoît DARTIES, Jérôme PALAYSI

LIRMM, Université Montpellier II - UMR 5506
161, rue Ada, 34392 Montpellier Cedex 5 - France

Résumé
Nous étudions deux problèmes algorithmiques inspirés des contraintes de routage rencontrées dans des réseaux sans-fil. Ces problèmes cherchent à satisfaire des requêtes de communication dans un environnement où deux nœuds ne sont pas nécessairement à portée directe d'émission et où les émissions trop proches génèrent des zones de brouillage. Pour satisfaire une requête il est possible de lui assigner une route dans le réseau que devra suivre le message de la requête. Pour éviter les brouillages nous pouvons envisager de temporiser l'émission de certains nœuds. Notre objectif est de minimiser le temps nécessaire pour satisfaire toutes les requêtes. Les deux problèmes diffèrent en ce que les routes données dans l'un (DAWN-paths) sont à trouver dans l'autre (DAWN-requests). Nous montrons en particulier que ces problèmes sont NP-difficiles et non approximables en général à un facteur constant près. Nous déterminons la frontière entre polynomialité et NP-complétude des problèmes de décision associés.

Mots-clés : réseaux radio, problème de routage, complexité algorithmique

1. Introduction

Nous nous intéressons à 2 problèmes algorithmiques inspirés du fonctionnement de réseaux sans-fil supervisés par un opérateur dans un modèle **half-duplex** (un nœud du réseau ne peut à la fois émettre un message et en recevoir un autre, mais il peut ignorer une réception pour procéder à une émission), **Δ-port-émission** (émission omnidirectionnelle : chaque message émis par un nœud atteint tous les nœuds voisins), **1-port-réception** (chaque nœud ne peut recevoir qu'un seul message à la fois). Une description de ce modèle est faite dans le chapitre 4 de la thèse de Guillaume Chelius [3] mais il apparaît déjà dans [4] pour des problèmes de broadcast. Récemment le problème du rassemblement (*gathering* en anglais) a été étudié dans ce même contexte [2, 1]. Les réseaux que nous considérons sont synchrones : le temps est divisé en étapes de longueurs égales (slots) et chaque message doit être intégralement émis et reçu dans un seul slot. Enfin la topologie des réseaux est fixée, au moins le temps pendant lequel les problèmes doivent être résolus. C'est dans ce contexte qu'émergent nos deux problèmes algorithmiques : satisfaire le plus rapidement possible une collection de **requêtes de communication** (une requête est un couple de nœuds du réseau) en indiquant aux nœuds du réseau les dates auxquelles ils doivent faire suivre les paquets qui transitent par eux.

Nous modélisons le réseau radio par un graphe non orienté $G = (V, E)$, où V représente l'ensemble des nœuds du réseau et E l'ensemble des paires de nœuds à portée d'émission l'un de l'autre. Un **parcours** dans un graphe G est une liste ordonnée de sommets du graphe, telle que 2 sommets consécutifs dans la liste sont adjacents dans G. Nous considérerons seulement des **parcours simples** pour lesquels le nombre d'occurences d'un sommet dans la liste est au plus 1. Nous utilisons un parcours pour modéliser une route de communication dans le réseau. La **taille d'un parcours** est égale à son nombre de sommets moins 1. Un graphe G et une collection de requêtes étant donnés, nous appelons **fonction de routage** une fonction P qui à toute requête $r = (s, t)$ associe un parcours $P(r)$ (encore noté P_r) de G commençant et terminant respectivement par s et t. Un graphe G, une collection de requêtes R et une fonction de routage étant donnés, une **assignation de dates** (ou fonction de datage) est une fonction qui, à tout couple (r, x) où $r \in R$ et $x \in P(r)$ avec $x \neq p$ [1]associe un entier naturel positif : la date à laquelle le sommet x doit relayer le message de la requête r. Une assignation de dates d est **correcte** si et seulement si pour toute requête r avec $P_r = (x_0, x_1, ..., x_k)$ on a $d(r, x_0) < d(r, x_1) < ... < d(r, x_k)$. Nous dirons qu'elle est de plus **sans conflit** si et seulement si lorsque $d(r, x_i) = d(r', y_j)$ les conditions suivantes sont satisfaites :

1. $x_i \neq y_j$ (pas de multiplexage, un nœud n'émet qu'un seul message à la fois)
2. $x_{i+1} \neq y_j$ et $y_{j+1} \neq x_i$ (half-duplex)
3. $\{x_i, y_{j+1}\} \notin E(G)$ et $\{y_j, x_{i+1}\} \notin E(G)$ (Δ-port-émission et 1-port-réception)

Compte tenu d'une collection de requêtes de communication à satisfaire dans un réseau radio synchrone, le problème DAWN (*Date Assignment in Wireless Network*) se propose de trouver une assignation de dates correcte et sans conflit le long de routes de communication. Selon si les routes de communication à suivre sont imposées (données par des tables de routage) ou pas, nous distinguerons respectivement deux types de problèmes : DAWN-paths et DAWN-requests. Chacun de ces problèmes fait l'objet d'une des deux sections suivantes.

Nous adoptons la terminologie définie dans [5] : un problème est dit **non approximable** à un facteur constant près s'il n'existe pas d'algorithme d'approximation polynomial avec une garantie de performance qui est une constante. Par ailleurs un algorithme d'approximation possède une garantie de performance qui est une constante (disons ρ) si pour toute instance du problème on a $\frac{C}{C^*} \leq \rho$ où C est le coût de la solution donnée par l'algorithme et C^* le coût de la solution minimum (nos problèmes sont des problèmes de minimisation).

Dans les 2 sections suivantes nous montrons que ces problèmes sont en général NP-difficiles et non approximables à un facteur constant près. Ces résultats concernant la NP-difficulté et la non approximabilité reposent sur des résultats connus pour le problème de coloration de graphes. Pour tout entier naturel D le problème D-COLORATION demande s'il est possible d'affecter un entier de l'intervalle $[1, D]$ à chaque sommet d'un graphe non orienté donné de sorte que deux sommets adjacents n'aient pas le même entier assigné. Pour $D \geq 3$, le problème D-COLORATION est NP-complet. Le problème de minimisation associé est min-COLORATION, connu pour être NP-difficile et non-approximable à un facteur constant près.

Pour les deux problèmes nous donnons toutefois un algorithme dont la complexité est exponentielle uniquement en fonction du nombre de requêtes à satisfaire (mais pas dans la taille du graphe).

Enfin, lorsqu'il s'agit de déterminer si une instance est réalisable en au plus D étapes (D entier naturel) alors DAWN-requests s'avère à peine plus ardu : la frontière entre polynomialité et NP-complétude se trouve entre 2 et 3 pour DAWN-paths mais entre 1 et 2 pour DAWN-requests.

[1] le dernier nœud du parcours, destinataire du message, n'a pas à le relayer.

2. Le problème DAWN-paths

Dans cette section nous étudions le problème DAWN-paths.

Donnée	:	Un graphe non orienté G, une collection de requêtes $(r_i = (s_i, t_i))_{1 \leq i \leq K}$, un entier naturel D, une fonction de routage P qui associe à chaque requête r_i un parcours $P(r_i)$ reliant les sommets de r_i.
Question	:	Existe-t-il une assignation de dates correcte et sans conflit, telle que la plus grande date assignée soit inférieure ou égale à D ?

Problème de décision 1: Le problème DAWN-paths

Nous appelons « min-DAWN-paths » la version optimisation de ce problème qui consiste à trouver une assignation de dates minimisant la plus grande date assignée D. La figure 1 illustre ces problèmes sur un exemple.

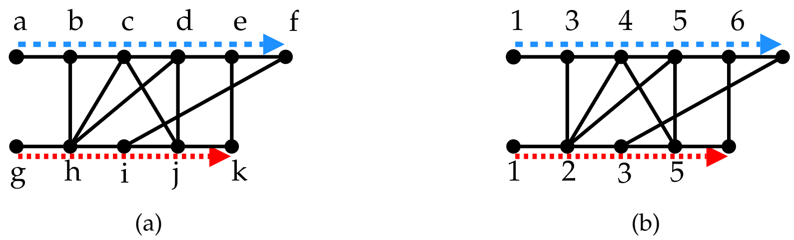

FIG. 1 – (a) une instance (G, R, P) de min-DAWN-paths avec 2 requêtes $r_1 = (A, F)$, $r_2 = (G, K)$, $P(r_1) = (A, B, C, D, E, F)$ et $P(r_2) = (G, H, I, J, K)$. La figure (b) montre une assignation de dates correcte, sans conflit et minimum. Sur cette instance, l'approches gloutonne consistant à privilégier la requête dont la destination est la plus loin à atteindre, ne donne pas une solution optimale.

Nous montrons dans la sous-section suivante que min-DAWN-paths est NP-Difficile en général et non approximable à un facteur constant près. Nous montrons ensuite que DAWN-paths est NP-complet, même si on se restreint à des instances où le temps maximum donné est supérieur ou égal à 3. Par contre dans la deuxième sous-section, nous prouvons que le calcul d'une solution s'il en existe une en 2 étapes au plus peut se faire en temps polynomial. Nous donnons pour terminer un algorithme exponentiel seulement en fonction du nombre de requêtes).

2.1. Un problème difficile

Pour tout entier naturel D, nous définissons le problème D-DAWN-paths depuis le problème général DAWN-paths en supprimant de la donnée l'entier naturel D. Nous montrons alors que pour tout entier naturel $D \geq 3$ le problème D-DAWN-paths est NP-complet (ce qui implique naturellement la NP-complétude du problème DAWN-paths et la NP-Difficulté de min-

DAWN-paths). Nous montrons par ailleurs qu'il n'existe pas, sauf si $P = NP$, d'algorithme polynomial d'approximation pour le problème min-DAWN-paths.

Théorème 1 *Le problème min-DAWN-paths est NP-Difficile et non approximable à une constante multiplicative près, même restreint à des parcours de taille* 1*. Pour tout entier naturel* $D \geq 3$*, le problème de décision D-DAWN-paths est NP-Complet même restreint à des parcours de taille* 1*.*

Preuve:

Soient Z et T deux algorithmes polynomiaux dont la donnée est un graphe non orienté G quelconque. Le premier construit un graphe $Z(G) = G'$ avec $V(G') = \{(1,x)|x \in V(G)\} \cup \{(2,x)|x \in V(G)\}$ et $E(G') = \big\{\{s_x, t_x\}|x \in V(G)\big\} \cup \big\{\{s_x, t_y\}|\{x,y\} \in E(G)\big\}$, en notant, par souci de clarté, s_x le couple $(1,x)$ et t_x le couple $(2,x)$ pour tout sommet x de $V(G)$. Le deuxième algorithme T construit l'ensemble de requêtes $T(G) = \{r_x = (s_x, t_x)|x \in V(G)\}$. Une fonction de routage F associe à chaque requête r_x le parcours $p_x = (s_x, t_x)$.
Soit c une coloration sans conflit des sommets de G. Soit d l'assignation de dates qui à tout couple (r_x, s_x) associe la date $c(x)$. Alors d est une assignation de dates correcte (évident) et sans conflit (puisque si $\{s_x, t_y\}$ est une arête de G' alors $\{x, y\}$ est une arête de G et donc $c(x) \neq c(y)$ et par conséquent $d(r_x, s_x) \neq d(r_y, s_y)$). Notons que $max(c) = max(d)$.
Soit d une assignation, correcte et sans conflit, de dates pour l'instance $\big(Z(G), T(G), F\big)$. Soit c la fonction qui à tout sommet x de G associe la couleur $d(r_x, s_x)$. Notons que $max(d) = max(c)$. La coloration obtenue est sans conflit puisque si x et y sont voisins dans G alors par construction deux arêtes $\{r_x, t_y\}$ et $\{r_y, t_x\}$ existent dans G' et donc $d(r_x, s_x) \neq d(r_y, s_y)$.
Nous en déduisons qu'à toute coloration c de G correspond une assignation de dates correcte et sans conflit de $(Z(T), T(G), F)$ telle que $max(c) = max(d)$. La réciproque est également vraie et donc : puisque min-COLORATION est NP-difficile et non approximable il en est de même pour min-DAWN-paths, et puisque D-COLORATION est NP-complet et que D-DAWN-paths est un problème dans NP alors D-DAWN-paths est NP-complet.

□

2.2. Cas polynomiaux

Dans cette sous-section nous mettons en évidence les cas polynomiaux. Pour tout entier naturel K, le problème d'optimisation min-DAWN-K-paths consiste à minimiser le nombre d'étapes nécessaires pour satisfaire K requêtes.

Théorème 2 *Pour tout entier naturel* K*, le problème min-DAWN-*K*-paths est polynomial.*

Preuve:

Pour un entier K donné, considérons (G, R, P) une instance du problème min-DAWN-K-paths ; respectivement un graphe, une collection de requêtes $r_1 \dots r_K$ et une fonction de routage. Soit V' l'ensemble des K-uplets $(e_1 \dots e_K)$ où e_i est un sommet du parcours associé par P à la requête r_i. Chacun de ces K-uplets représente un état possible du réseau : la composante e_i indiquant la position du message de la requête r_i sur son parcours. Deux états $(e_1, \dots, e_k)$ et $(f_1, \dots, f_k)$ sont compatibles s'il est possible de passer du premier au deuxième en une étape, en faisant émettre simulanément l'ensemble des nœuds $\{e_i | e_i \neq f_i, 1 \leq i \leq K\}$. Le K-uplet $(s_1, \dots s_i, \dots s_k)$ représente l'état initial du réseau, et le K-uplet $(t_1, \dots t_i, \dots t_k)$ l'état final du réseau.
Soit E' l'ensemble de tous les couples (x, y) où x et y sont des états compatibles de V'. Clairement (V', E') est un graphe orienté constructible en temps polynomial. Nous pouvons associer

à tout chemin Q entre l'état $(s_1, \dots s_i, \dots s_k)$ et l'état $(t_1, \dots t_i, \dots t_k)$ une assignation de dates d, correcte et sans conflit, telle que la plus grande date assignée est égale au nombre d'arcs du chemin, et réciproquement. Précisément, soient S et T deux sommets consécutifs dans Q. Si le i-ème sommet x de S est différent du i-ème sommet de T alors $d(r_i, x)$ est défini égal au rang de S dans le chemin Q.
Puisque la recherche d'un plus court chemin dans un graphe est un problème bien connu pour être polynomial nous pouvons directement conclure. □

Le prochain théorème montre qu'il est possible de vérifier en temps polynomial si une instance peut être satisfaite en 2 étapes au plus. Nous ne pouvons espérer mieux puisque le théorème 1 affirme que le problème devient NP-complet à partir de 3 étapes.
La preuve utilise un résultat connu du problème de coloration sur listes. Ce problème est défini par la donnée d'un graphe $G = (V, E)$ non orienté, d'un ensemble de couleurs C (par exemple un intervalle d'entiers) et d'une fonction L de V dans $P(C)$ (l'ensemble des parties de C). Le but est de trouver une fonction c de V dans C de telle sorte que pour chaque sommet ν du graphe, $c(\nu)$ appartienne à $L(\nu)$ et soit différent de chaque couleur choisie pour ses voisins. Lorsqu'on se restreint aux instances dont les listes sont de longueur au plus 2 ce problème est connu pour être polynomial.

Théorème 3 *Le problème de décision 2-DAWN-paths est polynomial*

Preuve:

Soit $(G, R = r_1, \dots r_K, P)$ une instance du problème 2-DAWN-paths. Nous pouvons supposer que pour chaque requête r le parcours $P(r)$ est de la forme (s, t) ou (s, l, t), où l est adjacent à s et t dans G, puisque s'il existe un parcours plus long, l'instance est clairement sans solution.
Soit G' le graphe non orienté avec $V(G') = \{(r, x) | r \in R, x \in P(r)\}$ avec $x \neq t$. Il existe dans G' une arête entre deux sommets (u, x) et (ν, y) si et seulement si pour toute assignation de dates sans conflit $d(u, x) \neq d(\nu, y)$.
Soit L une assignation de listes de couleurs aux sommets de G'. Pour toute requête r telle que $P(r)$ contient 3 sommets (s, l, t) nous assignons dans G' la liste (1) au sommet (r, s) et la liste (2) au sommet (r, l). Nous assignons la liste de couleurs $(1, 2)$ pour les autres sommets du graphe G'.
S'il existe une solution c au problème de coloration sur listes pour l'instance (G', L) alors il existe une assignation de dates d correcte, sans conflit et en 2 étapes au plus : il suffit de poser $d(r, x) = c(r, x)$. La réciproque est également vraie.
Rappelons pour conclure que le problème de coloration sur listes est polynomial lorsque les listes sont de longueur au plus 2.

□

3. Le problème DAWN-requests

Dans cette section nous étudions le problème DAWN-requests.
Nous appelons « min-DAWN-requests » la version optimisation de ce problème qui consiste à trouver une assignation de dates minimisant la plus grande date assignée D.
Nous montrons dans la sous-section suivante que min-DAWN-requests est NP-Difficile en général et non approximable à une constante multiplicative près. Par contre, à l'instar de min-

Donnée	:	Un graphe non orienté G, une collection de requêtes $(r_i = (s_i, t_i))_{1 \leq i \leq K}$, un entier naturel D.
Question	:	Existe-t-il une fonction de routage P qui associe à chaque requête r_i un parcours $P(r_i)$ reliant les sommets de r_i, et une assignation de dates correcte et sans conflit, telle que la plus grande date assignée soit inférieure ou égale à D ?

Problème de décision 2: Le problème DAWN-requests

DAWN-paths, il devient raisonnable de penser calculer une solution lorsque le nombre de requêtes est suffisamment petit.

3.1. Un problème difficile

Pour tout entier naturel D, nous définisson le problème D-DAWN-requests à partir du problème plus général DAWN-requests en supprimant de la donnée l'entier naturel D. Nous montrons dans cette section que pour tout entier naturel $D \geq 2$ le problème D-DAWN-requests est NP-complet (ce qui implique naturellement la NP-complétude du problème DAWN-requests et la NP-Difficulté de min-DAWN-requests). Nous montrons par ailleurs qu'il n'existe pas, sauf si $P = NP$, d'algorithme polynomial d'approximation pour le problème min-DAWN-requests.

Théorème 4 *Le problème d'optimisation min-DAWN-requests est NP-Difficile et non approximable à une constante multiplicative près. Pour tout entier naturel* $D \geq 3$*, le problème de décision D-DAWN-requests est NP-Complet*

Preuve:

Soit Z et T les algorithmes définis dans la preuve du théorème 1. Soit G une instance du problème min-COLORATION. Soit G' le graphe $Z(G)$, et G'' le graphe avec $V(G'') = V(G')$ et $E(G'') = E(G') \cup \{\{t_x, t_y\} | x, y \in V(G)\}$. Considérons que $(G'', T(G))$ est une instance de min-DAWN-requests.
Soit S une solution à $(G'', T(G))$ de coût z. Supposons qu'il existe dans S une requête $r_i = (s_i, t_i)$ telle que le parcours $P(r_i)$ proposé contient un sommet t_j, $j \neq i$, auquel la date d a été assignée. Alors notre construction implique que la date d ne peut être attribuée à aucun élément s_k de n'importe quel autre parcours. Nous pouvons alors nous ramener à une solution S' de coût $z' \leq z$ dans laquelle $P(r_i)$ peut être réduit à un seul élément (s_i) auquel nous attribuons la date d. Ainsi à partir d'une solution S de coût z, nous calculons une solution S^*, dite *solution propre*, de coût au plus égal à z, et telle qu'aucun parcours ne contient de sommet t.
Comme pour la preuve du théorème 1, on peut déduire de toute coloration c de G une solution propre dans G'' de même coût, et réciproquement. Puisque min-COLORATION est NP-difficile et non approximable il en est de même pour min-DAWN-requests.
Pour n'importe quel entier D, la même construction permet de réduire D-COLORATION au problème D-DAWN-request. Puisque D-DAWN-requests est dans NP nous pouvons conclure qu'il est NP-complet.

□

Si la démonstration précédente permet de conclure que D-DAWN-requests est NP-Complet pour $D \geq 3$, le théorème suivant affirme également cette NP-Complétude pour $D = 2$.

Théorème 5 *Le problème de décision 2-DAWN-requests est NP-Complet*

Preuve:

Soit I une instance du problème 3-SAT construite sur un ensemble de variables X.
Pour chaque variable x de X soit H_x le biparti complet dont les sommets des deux partitions sont $\{(1, x), (1, \neg x)\}$ et $\{(2, x), (2, \neg x)\}$.
Pour chaque clause $C = \{l_1, l_2, l_3\}$ de I, soit F_C le biparti complet dont les deux partitions sont $\{(1, C), (2, C)\}$ et $\{(C, l_1), (C, l_2), (C, l_3)\}$.
Soit $I' = (G, R, 2)$ l'instance de DAWN-requests où G contient tous les sommets et les arêtes des graphes H_x et F_C pour toute variable x de X et toute clause C de I. De plus G contient les arêtes $\{(C, l), (2, l)\}$ pour tout litéral l et toute clause C tels que $l \in C$. La figure 2 présente un exemple de graphe G construit à partir d'une instance de 3-SAT. La collection R contient exactement toutes les requêtes de la forme $((1, C), (2, C))$ où C est une clause de I et toutes les requêtes de la forme $((1, l), (2, l))$ où l est un littéral d'une clause de I.
Supposons que I' admette une solution. Un littéral l de I étant donné, affectons-lui la valeur vrai si et seulement si $(1, l)$ émet à la date 1. Une clause C étant donnée, exactement un sommet du type (C, l) émet à l'étape 2. Ce sommet est relié à un sommet $(2, l)$ qui n'a pu recevoir le message qu'à l'étape 1. Donc l vaut vrai et C est satisfait.
Réciproquement, supposons que I admette une solution. Pour tout litéral l de I qui a pour valeur vrai, assignons la date 1 au sommet $(1, l)$ et la date 2 au sommet $(2, \neg l)$. Soit C une clause de I. La date 1 est assignée au sommet $(1, C)$. La date 2 doit être assignée à un des sommets voisins (et un seul). Nous pouvons choisir un des couples dont le littéral l vaut vrai. Cette date est possible puisque (C, l) est seulement relié à $(2, C)$ (destinataire) et $(2, l)$ qui a déjà reçu le message à la date 1.
Enfin, puisque 2-DAWN-request est dans NP, il s'agit bien d'un problème NP-complet.

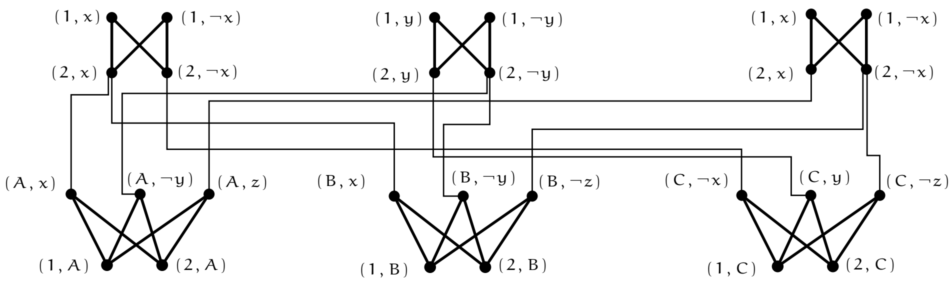

FIG. 2 – graphe construit à partir de l'instance $\{\{x, \neg y, z\}, \{x, \neg y, \neg z\}, \{\neg x, y, \neg z\}\}$

□

3.2. Cas polynomiaux

Pour tout entier naturel K, le problème min-DAWN-K-paths consiste à minimiser le nombre d'étapes nécessaires pour satisfaire K requêtes. Nous proposons le théorème suivant :

Théorème 6 *Le problème de décision DAWN-K-requests est polynomial*

Idée de preuve:

Reprenons la preuve de la polynomialité de DAWN-K-paths, présentée en sous-section 2.2. Les problèmes DAWN-K-paths et DAWN-K-requests sont très similaires, à l'unique différence que l'on ne sait pas dans le second par quels nœuds chaque requête transite. Ainsi dans l'algorithme de résolution polynomial de la preuve, nous ne générons pas seulement l'ensemble des k-uplets $(e_1 \dots e_K), \forall e_i \in P(r_i)$, mais l'ensemble des k-uplets $(e_1 \dots e_K), \forall e_i \in V(G)$. Le nombre d'états augmente mais reste polynomial en fonction de la donnée (n^K avec K constant). La suite de la preuve reste inchangée. □

4. Conclusion

Nous avons analysé la complexité du problème de satisfaction de requêtes dans un réseau radio synchrone. Le tableau 1 résume les résultats présentés dans cet article.

<table>
<tr><th colspan="2">Instance DAWN-paths :</th><th>Complexité :</th><th colspan="2">Instance DAWN-requests :</th></tr>
<tr><td colspan="2">Min-DAWN-paths</td><td>NP-Difficile, non approximable à une constante multiplicative</td><td colspan="2">Min-DAWN-requests</td></tr>
<tr><td rowspan="2">D-DAWN-paths</td><td>$d \leq 2$</td><td>Polynomial</td><td>$d \leq 1$</td><td rowspan="2">D-DAWN-requests</td></tr>
<tr><td>$d \geq 3$</td><td>NP-Complet</td><td>$d \geq 2$</td></tr>
<tr><td colspan="2">DAWN-K-paths</td><td>Polynomial (*algorithme en* $O(n^k)$)</td><td colspan="2">DAWN-K-requests</td></tr>
</table>

TAB. 1 – Tableau récapitulatif des résultats de complexité

Nous prévoyons d'analyser la complexité des problèmes DAWN-paths et DAWN-requests sur des topologies particulières. Cette étude permettra peut-être de découvrir des cas de polynomialité indépendamment du nombre de requêtes. La recherche d'heuristiques avec garantie de performance pour les instances difficiles constitue une suite naturelle de notre étude. Enfin, une variante de DAWN, dite consisterait à satisfaire un ensemble de requête à un instant donné, tandis que les émissions d'autres requêtes ont déjà été programmées précédemment et interdisent ainsi certains nœuds d'émettre à certaines étapes.

Bibliographie

1. J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Pérennes. Hardness and approximation of gathering in static radio networks. In *FAWN06, Pisa, Italy*, March 2006.
2. J.-C. Bermond and J. Peters. Efficient gathering in radio grids with interference. In *Septièmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications (AlgoTel'05)*, Presqu'île de Giens, May 2005.
3. G. Chelius. *Architectures et Communications dans les réseaux spontanés sans fil*. PhD thesis, INSA de Lyon, INRIA Rhône Alpes, France, April 2004.
4. I. Chlamtac and S. Kutten. On broadcasting in radio networks - Problem analysis and protocol design. *IEEE Transactions on Communications*, 33 :1240–1246, December 1985.
5. T. Cormen, C. Leiserson, and R. Rivest. *Introduction à l'algorithmique, 2nde édition*. Dunod, 2001.