



HAL
open science

Request Satisfaction Problem in Synchronous Radio Networks

Benoit Darties, Sylvain Durand, Jérôme Palaysi

► **To cite this version:**

Benoit Darties, Sylvain Durand, Jérôme Palaysi. Request Satisfaction Problem in Synchronous Radio Networks. ADHOC-NOW: Ad-hoc, Mobile and Wireless Networks, Sep 2008, Sophia Antipolis, France. pp.451-462, 10.1007/978-3-540-85209-4_36 . lirmm-00322362

HAL Id: lirmm-00322362

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00322362>

Submitted on 17 Sep 2008

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.

Request Satisfaction Problem in Synchronous Radio Networks

Benoît Darties, Sylvain Durand, and Jérôme Palaysi

LIRMM, Université Montpellier II
161 rue Ada, 34392 Montpellier Cedex 5 - France
{benoit.darties,sylvain.durand,jerome.palaysi}@lirmm.fr

Abstract. We study two algorithmical problems inspired from routing constraints in a multihop synchronous radio network. Our objective is to satisfy a given set of communication requests in the following model: nodes send omnidirectional radio transmissions in synchronous slots; during a given slot, a node can receive a message from an adjacent node if and only if no other neighbour is transmitting - otherwise, radio interferences may occur if two or more neighbors transmit in the same slot -. The objective is to minimize the number of slots used. The two problems differ in that the routing policy may be imposed (DAWN-path), or not (DAWN-request). In this second case, a path must be assigned for each request, to define the nodes to use to reach the destination from the source. We present some complexity results, in particular showing that both problems are NP-hard when the network is restricted to a tree. We also present a polynomial algorithm in $O(n^{2K})$ when the number of requests is bounded (by above) by a constant K .

Key words: Radio Network, Request Satisfaction, Complexity

1 Introduction

A radio network is a collection of transmitter-receiver stations (or nodes) communicating with one another via multihop wireless links. The use of the radio medium implies some restrictions and properties: whenever a node transmits, all the nodes in its communication range may receive the transmission. Incoming messages have to be forwarded to reach nodes which are located more than one hop away from the source. Since all nodes share the same frequency channel, a collision may occur if two or more neighbors transmit simultaneously, preventing correct reception of the message.

In this paper, we study two communication problems inspired from routing constraints in this kind of network. We consider the following simplified communication model, which has been widely used for the broadcast problem [1, 4–6, 10, 12, 13] or the gathering problem [2, 3] in multihop radio networks : nodes send messages in synchronous slots; during each slot each node acts either as a transmitter or a receiver. A node acting as a transmitter sends a message which can potentially reach the nodes that are in its communication range. A node

acting as a receiver successfully receives a message from a transmitter node if no other neighbor transmits in this slot. If two or more neighbors of a receiver node u transmit simultaneously in a given slot, then the messages may interfere with each others (collide) and the messages are not transmitted successfully to u . We note that two neighbors u and v may successfully transmit in the same slot to u' and v' , if we assume that u' and v are not adjacent, and respectively v and u' : the node u acting as transmitter simply ignores the transmission of v and reciprocally. Such a network is characterized as a Δ -port transmission, 1-port reception, half-duplex, synchronous network. We suppose that the network topology is fixed, at least during the time the problem must be solved. All those properties specify the model we use in this work.

In this context, we consider the problem of satisfying a set of communication requests within a minimum timeframe, by indicating for each node the slots on which it has to relay transiting packets. A request is a couple of source-destination nodes representing the starting and ending nodes of a given message. The second section of this paper details the model and introduces the DAWN-path and DAWN-request problems. General complexity results are discussed in a third section where we will show that these problems are quite difficult even for particular cases. However we present in the fourth section a polynomial algorithm when the number of requests is bounded by a constant K .

2 Describing the model and expressing the problem

2.1 The model

The network is represented as an undirected graph G where the set $V(G)$ of vertices corresponds to the set of nodes of the network. An edge $e = \{u, v\} \in E(G)$ denotes that u can directly communicate to v (no additional node is required to relay the message) and reciprocally¹.

A *request* r is a couple $(s, t) | s, t \in V(G)$, where s represents the source and t the destination of the request.

A *path* of length k in a graph G is an ordered list (v_0, v_1, \dots, v_k) , where $v_i \in V(G)$ for any $i \in [0, k]$, and such that the edge (v_i, v_{i+1}) exists in $E(G)$ for any $i \in [0, k - 1]$, and all the edges are different. Throughout the paper all the considered paths are simple paths, that is, paths which visit a vertex at most once. Paths are used here to represent a communication road in the network.

Given a graph G and a collection of communication requests R , let P be a *routing function* on R which associates to each $r = (s, t) \in R$ a path $P(r)$ in G , (also denoted by P_r), beginning with s and ending with t .

Given a graph G , a collection of requests R , and a routing function P , we define a *date assignment* d to be a function which takes two arguments r and x , with $r = (s, t) \in R$, $x \in P_r$ and $x \neq t$, and returns an integer $d(r, x)$. This

¹ We consider that if x can directly communicate with y then y can directly communicate with x . We can deduce that the graph is symmetrically directed, and will be represented by an undirected graph.

integer corresponds to a slot, such that x transmits the message of r to the next hop during this slot. Multiplexing is not allowed, this implies that each transmission only contains a single message. A date assignment d is said to be **valid** if and only if for each request $r = (x_0, x_k)$ with $P_r = (x_0, x_1, \dots, x_k)$ the proposition $d(r, x_0) < d(r, x_1) < \dots < d(r, x_{k-1})$ is true. Moreover we say that a valid assignment is *conflict-free* if and only if for each $d(r, x_i) = d(r', y_j)$ where $r \neq r'$, the following holds :

1. $x_i \neq y_j$: prevents multiplexing
2. $x_{i+1} \neq y_j \wedge y_{j+1} \neq x_i$: a node cannot receive and transmit simultaneously
3. $\{x_i, y_{j+1}\} \notin E(G) \wedge \{y_j, x_{i+1}\} \notin E(G)$: Δ -port-transmission, 1-port-reception

We note $\max(d) = \max_{r \in R, x \in V(G)} d(r, x)$ the cost of d , i.e. the number of slots used by a date assignment d .

2.2 The DAWN problem

Given a set of requests to satisfy in a synchronous radio network and a maximum number of slots, the problem DAWN (*Date Assignment in Wireless Network*) consists in finding a conflict-free date assignment along communication paths. According to whether the paths are given (e.g. by the routing function) or not, we distinct two main problems: DAWN-paths and DAWN-request.

The *DAWN-path* problem is stated as follows:

INPUT: An undirected graph G , a collection of requests $R = \{r_i = (s_i, t_i) | 1 \leq i \leq K\}$, a routing function P on R which associates to each request r_i a path $P(r_i)$ linking the vertices of r_i , a natural integer D .

QUESTION: Does a valid and conflict-free date assignment exist in such a manner that the number of required slots is lower than or equal to D ?

Let min-DAWN-path be the optimization version of DAWN-path. For each natural integer D we define the D -DAWN-path problem as the subclass of DAWN-path where the maximum number of allowed slots is D . Note that D is bounded by above by $|V(G)| \times |R|$ otherwise the answer is obviously "yes". Figure 1 presents an instance of DAWN-path (1(a)) and a solution (conflict-free assignment) to it (1(b)).

We observe that there is no optimal fixed routing for the DAWN-path problem [8, page 97]. This leads us to propose the *DAWN-request* problem:

INPUT: An undirected graph G , a collection of requests $R = \{r_i = (s_i, t_i) | 1 \leq i \leq K\}$, a natural integer D .

QUESTION: Does a valid and conflict-free date assignment exist in such a manner that the number of required slots is lower than or equal to D ?

As stated before, we call min-DAWN-request the optimization version of DAWN-request, and D -DAWN-request the subclass of DAWN-request where D is the maximum number of allowed slots.



Fig. 1. An instance (G, R, P) of min-DAWN-path containing 2 requests $r_1 = (a, f)$, $r_2 = (g, k)$, $P(r_1) = (a, b, c, d, e, f)$ and $P(r_2) = (g, h, i, j, k)$ (sub-fig. a). A valid and conflict-free date assignment and within a minimum number of slots (sub-fig. b).

3 Complexity results

We adopt the terminology of [7]: a problem is not approximable within a constant factor if no polynomial approximation algorithm with a constant performance guarantee exists. Moreover an approximation algorithm has a constant performance guarantee of ρ if for each instance I of a problem it finds a solution the cost of which is at most ρ times the cost of the optimal solution for instance I .

In the following subsection we show that in general min-DAWN-path and min-DAWN-request are NP-hard and not approximable within a constant factor. These results are based on the complexity of coloring problems on graphs. The D-COLORING problem [11] consists in assigning a color (represented by a number bounded by above by D) to each vertex assuming that two adjacent vertices are assigned different colors. It is known that D-COLORING is NP-complete for any constant $D \geq 3$, and that the corresponding minimization problem min-COLORING is NP-hard and not approximable within a constant factor.

In a second subsection we show these problems remain NP-hard even when the network is a tree, but here the reduction does not enable us to prove the inapproximability (within some constant). In the third subsection we locate the boundaries between polynomiality and NP-completeness for D -DAWN-path and D -DAWN-request when only D varies.

3.1 Two difficult problems

The first theorem proves the NP-completeness of both problems in general.

Theorem 1. *Problems min-DAWN-path and min-DAWN-requests are NP-hard and not approximable within some constant factor. For any $D \geq 3$, decision problems D -DAWN-path and D -DAWN-request are NP-complete.*

Proof. We first prove the NP-completeness of D-DAWN-request by a reduction to D-COLORING.

D-DAWN-request is in NP : given a routing function P and a date assignment d as a solution of an instance, one can check in a polynomial time if P enables

each message to reach their destination, and if d is a valid conflict-free date assignment using fewer than D slots.

Let $I_C = (G_C)$ be an instance of D-COLORING, and let us note $n = |V(G_C)|$. From I_C we define an instance $I = (G, R)$ of D-DAWN-request, where G is a graph such that $V(G) = \{s_x, t_x | x \in V(G_C)\}$ and $E(G) = \{\{s_x, t_x\} | x \in V(G_C)\} \cup \{\{s_x, t_y\} | \{x, y\} \in E(G_C)\} \cup \{\{t_x, t_y\} | x, y \in V(G_C)\}$. We define the set $R = (r_x = (s_x, t_x) | x \in V(G_C))$ of n communication requests. Clearly, the instance I can be constructed in a polynomial time. Figure 2(b) gives an example of a graph G constructed from the graph G_C of figure 2(a).

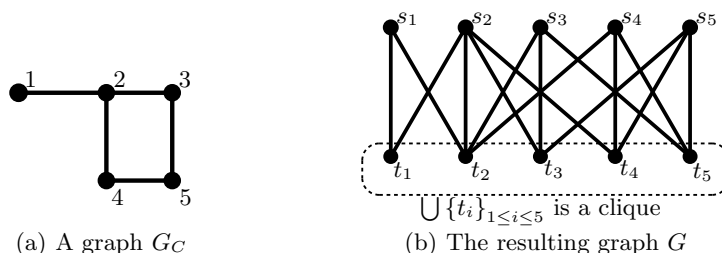


Fig. 2. Construction of G from G_c

We show that, if there exists a valid conflict-free date assignment for I using $k \leq D$ slots, then there exists a solution to the instance I_C of D-COLORING with cost $k \leq D$ and reciprocally.

Let $S = (P, d)$ be a solution of $I = (G, R)$ where P is a routing function for R and d a valid and conflict-free date assignment for (G, R, P) with cost $k = \max(d) \leq D$. Let us suppose there exists a request $r_i = (s_i, t_i)$ in S such that the message is not directly emitted from s_i to t_i , but requires at least one relay node $t_j | j \neq i$. If t_j transmits in slot u , then no other node s_l may transmit in the same slot, while $\cup_{i \leq n} t_i$ is a clique. Then we can extract a solution S' from S with cost $z' \leq z$ in which s_i transmits directly to t_i at slot u . Thus from any solution S with cost k , we compute a *proper* solution $S' = (P', d')$ with cost $k^* \leq k \leq D$ such that each message is directly transmitted from its source to its destination. Clearly $P'(r_i) = (s_i, t_i) \forall r_i \in R$. Let c be the function which assigns to each vertex $x \in V(G_C)$ the color $d'(r_x, s_x)$. Let us note that $\max(d') = \max(c) \leq D$. The resulting coloring is valid because if x and y are adjacent in G_C then by construction the edges $\{r_x, t_y\}$ and $\{r_y, t_x\}$ exist in G and imply that $d(r_x, s_x) \neq d(r_y, s_y)$.

Reciprocally, let c be a vertex coloring of G_C with cost $k \leq D$. Let P the routing function such that $P(r_i) = (s_i, t_i) \forall r_i \in R$, and d be the date assignment which associates the date $c(x)$ to each couple (r_x, s_x) . Then d is a valid conflict-free date assignment such that $\max(c) = \max(d)$.

To conclude, we claim that to any vertex coloring c of I_C corresponds a solution to I composed of a routing function P a valid and conflict-free date assignment d of (G, R) such that $\max(c) = \max(d)$, and reciprocally. Since D-COLORING is NP-complete for any $D \geq 3$ [11] and D-DAWN-request belongs to NP, then D-DAWN-request is also NP-complete for any $D \geq 3$. This proof can be extended to prove the NP-completeness of D-DAWN-path for any $D \geq 3$, by adding to the instance I the routing function P such that $P(r_i) = (s_i, t_i) \forall r_i \in R$. By adapting this proof to the optimization versions of these problems we show that min-DAWN-path and min-DAWN-request are NP-hard by a reduction to min-COLORING. Therefore the reduction preserves the inapproximability of min-COLORING, which is NP-hard and not approximable within some constant factor. This allows to conclude. \square

We now show that DAWN-path and DAWN-request are still NP-complete even if the network is a tree. This resolves an open question suggested in [9]. The proof can be extended to binary tree or Unit Disk Graph (intersection graph of disks with equal diameters). UDG are often used to model the topology of ad-hoc wireless communication networks. Let us introduce the following propositions:

Lemma 1. *Let $I = (G, R, P, D)$ be an instance of DAWN-path, let x a vertex from $V(G)$ and i an integer. Then there exists an instance $I' = (G', R', P', D)$ of the same problem with $V(G) \subseteq V(G')$, $R \subseteq R'$, such that :*

- each valid and conflict-free date assignment d on I' requires exactly D slots, and is also a valid and conflict-free date assignment on I ,
- for each valid and conflict-free date assignment d on I' and each request $r \in R'$ we have $d(r, x) \neq i$,
- the instance I can be constructed in a polynomial time.

Proof. Let us consider the following items:

- an instance $I = (G, R, P, D)$ of DAWN-path,
- a request $r \in R$ such that $P(r) = (s, \dots, x, y, \dots, t)$,
- a chain $C = \{c_1, c_2, c_3, \dots, c_{D+1}\}$ of length $D + 1$, with $V(G) \cap V(C) = \emptyset$,
- a request $r' = (c_1, c_{D+1})$,
- a natural integer $i \in [1, D]$.

We note $H = (V(G) \cup V(C), E(G) \cup E(C) \cup \{\{y, c_i\}\})$ and define $P(r') = (c_1, c_2, c_3, \dots, c_{D+1})$. Note that if G is a tree, then H is also a tree. Figure 3 schematically illustrates such a construction. We claim the following :

1. The instance $I' = (H, R \cup \{r'\}, P, D)$ can be constructed in polynomial time from $I = (G, R, P, D)$,
2. let d be a valid and conflict-free date assignment for I' , then we have $d(r', c_i) = i$, $d(r, x) \neq i$, and d is a valid date assignment for I . \square

This construction presented in the proof of lemma 1 will be used in the proof of theorem 2 to prevent some nodes from transmitting during certain slots. In

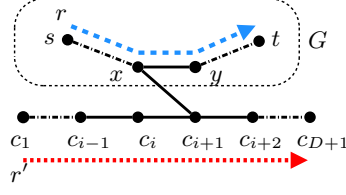


Fig. 3. How to prevent a node x from transmitting in slot i

the following, we say that a request $r = (s, t)$ starts at slot t if the source node s proceeds to the transmission of the message of r during the t th time slot.

Let u and D be two natural integers such that $D \geq 6$. We define a tight (u, D) DAWN-path instance as the DAWN-path instance $(C_{[1, D+7u]}, R, P, D)$ where $C_{[1, D+7u]}$ is a chain having $D + (7u)$ vertices $\{1, 2, \dots, D + 7u\}$ and the edges $\{i, i+1\} | \forall 0 \leq i \leq D + 7u - 1$. R is the set of $2u$ requests $\{r_1, \bar{r}_1, r_2, \bar{r}_2, \dots, r_u, \bar{r}_u\}$ with $r_i = \{(7i - 5, 7i + D - 9)\}$ and $\bar{r}_i = \{(7i - 6, 7i + D - 8)\} | \forall i \leq u$.

Lemma 2. *Let us consider a tight (u, D) instance for some integer u and D . Let us suppose d is a valid and conflict-free date assignment, and $i \in [1, u]$. We make the following observations:*

1. $(d(r_i, 7i - 5), d(\bar{r}_i, 7i - 6)) \in \{(5, 1), (1, 3)\}$,
2. $d(r_i, j + 1) = d(r_i, j) + 1, \forall j \in P(r_i) - \{7i + D - 9\}$,
3. $d(\bar{r}_i, j + 1) = d(\bar{r}_i, j) + 1, \forall j \in P(\bar{r}_i) - \{7i + D - 8\}$.

Given two natural integers $i \in [1, u]$ and $j \in P(r_i)$:

1. if $d(r_i, 7i - 5) = 1$ then $d(r_i, j) = j - 7i + 6$ and $d(\bar{r}_i, j) = j - 7i + 9$
2. if $d(r_i, 7i - 5) = 5$ then $d(r_i, j) = j - 7i + 10$ and $d(\bar{r}_i, j) = j - 7i + 7$

Proof. This obvious proof is left to the reader. □

Theorem 2. *DAWN-path and DAWN-request remain NP-complete even if the graph representing the network topology is a tree.*

Proof. The proof is based on a polynomial reduction of any instance of 3-SAT problem [11] to an instance (G, R, P, D) of DAWN-path where G is a tree.

Let $I_{SAT} = (U, W)$ be an instance of 3-SAT, composed of a set of variables $U = \{x_1, x_2, \dots, x_n\}$ and a set of clauses of 3 literals $W = \{c_1, c_2, \dots, c_m\}$. We note $n = |U|$, $m = |W|$, and set $D = m + 7n + 3$.

Let us consider the tight (n, D) instance (G_1, R_1, P, D) . The main idea of the proof consists in assigning two requests r_i and \bar{r}_i to each variable $x_i \in W$. Thus for the sake of clarity we note r_{x_i} the request r_i and $r_{\bar{x}_i}$ the request \bar{r}_i .

Let G_2 be the tree such that $V(G_2) = V(G_1) \cup (c_i, 1), (c_i, 2) | i \in [1, m]$ and $E(G_2) = E(G_1) \cup \{(c_i, 1), (c_i, 2)\}, \{7n + i, c_i^s\} | i \in [1, m]\}$. In the following we note c_i^s the couple $(c_i, 1)$ and c_i^t the couple $(c_i, 2)$ for any integer i . Let R_2

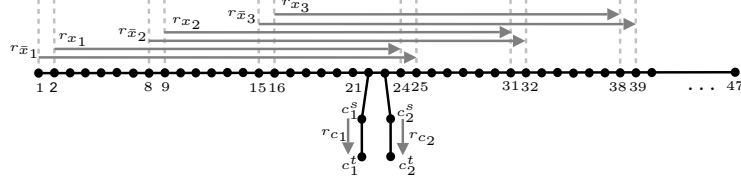


Fig. 4. A graph G_2 and a set of requests $R_1 \cup R_2$ constructed from an instance $I_{SAT} = (U, W)$ where $U = \{x_1, x_2, x_3\}$ and $W = \{c_1, c_2\}$. Here $D = 26$.

be the set of requests $\{r_{c_i} = (c_i^s, c_i^t) \mid i \in [1, m]\}$. Now, consider the instance $(G_2, R_1 \cup R_2, P, D)$ (see Figure 4 for an example).

By applying the construction of lemma 3.1 several times, we create an instance $I = (G, R, P, D)$ by adding elements to $(G_2, R_1 \cup R_2, P, D)$ as follows : for each request $r_{c_i} = (c_i^s, c_i^t)$ we prevent c_i^s from transmitting to c_i^t the message of request r_{c_i} during each slot $t \leq D$, except for 3 specific ones which are defined from the literals of the clause c_i : if c_i contains the positive (resp. negative) literal associated to the variable $x_j \mid j \leq n$, then c_i^s is allowed to transmit in slot $7n + i - 7j + 6$ (resp. $7n + i - 7j + 8$). Since G_2 is a tree, G is also a tree and the routing function is still obvious.

The size of I is polynomial in the size of I_{SAT} , and it can be constructed in a polynomial time. We claim that if there is a solution in D slots to I , then we can deduce a solution to the instance I_{SAT} and reciprocally.

Let us consider a valid and conflict-free date assignment d on the instance I . According to lemma 2 and for each integer $i \in [1, n]$, one request of $\{r_{x_i}, r_{\bar{x}_i}\}$ must start at slot 1, and the other as soon as possible, and once a request has been started, its progression cannot be stopped. Moreover each request $r_{c_i} \mid i \in [1, m]$ is clearly satisfied by d . The slot which has been assigned to (r_{c_i}, c_i^s) is one of the three allowable values defined from the literals of clause c_i . Our construction implies that there exists j such that $d(r_{c_i}, c_i^s)$ is of the form $7n + i - 7j + 6$ or $7n + i - 7j + 8$, according to x_j is a positive or a negative literal. If x_j is a positive literal, then the source of r_{x_j} is located on vertex $7j - 5$, i.e. at distance $7n + i_7j + 5$ from the vertex $7n + j$ which is adjacent to c_i^s . Then r_{x_j} necessarily starts at slot 1 - and we have $d(r_{x_j}, 7n + i) = d(r_{c_i}, c_i^s)$ - otherwise messages would collide. We adopt a similar reasoning when x_j is a negative literal. Then for each clause c_i , there exists at least one (positive or negative) literal $l \in c_i$, such that the request r_l starts before $r_{\bar{l}}$. By affecting the value "True" to all variables x_i where r_{x_i} has been started at slot 1 (i.e. before $r_{\bar{x}_i}$) and "False" otherwise, we obtain a solution to the instance I_{SAT} .

Reciprocally we can deduce a solution to the instance I from a solution to I_{SAT} : for each variable x_i we start the request r_{x_i} before $r_{\bar{x}_i}$ if and only if x_i has the value "True". For each clause c_i , we start r_{c_i} at the first valid and available slot (this slot exists since the clause c_i is satisfied by at least one literal).

To conclude we point out that 3-SAT is NP-complete and that DAWN-path belongs to NP. Then the DAWN-path is NP-complete even if the network is a tree. This implies the NP-completeness of DAWN-request, since there is one unique path linking each source to its destination when the network is a tree. \square

3.2 Locating the boundaries between polynomiality and NP-completeness

We have shown that DAWN-path and DAWN-request are NP-complete even when the network topology is very restrictive. In the following subsection, we focus our interest on the influence of the maximum number of slots D on the complexity of these problems.

Theorem 1 already affirms that D-DAWN-path and D-DAWN-request are NP-complete when $D \geq 3$. By the way when $D = 1$ one can verify in polynomial time if an instance can be satisfied: for each request $r = (s, t)$, s must be adjacent to t , and can only emit at the first slot. Hence t cannot be adjacent to another source node s_2 , and s must be a source only for t . The next theorem states that one can check in a polynomial time if there is a solution to a D-DAWN-path instance when $D \leq 2$.

Theorem 3. *The 2-DAWN-path decision problem is polynomial*

Proof. Let $I = (G, R, P)$ an instance of 2-DAWN-path. One can suppose that for each request $r = (s, t)$ the path $P(r)$ contains at most one vertex l between s and t , otherwise the instance is clearly insolvable. We propose an algorithm in three steps :

- Dates are forced for transmitters belonging to two-hop requests.
- Therefore dates 1 (resp. 2) are spread to every transmitter which cannot transmit during the second (resp. first) slot.
- Remaining dates are computed using a 2-SAT-like algorithm. \square

Theorems 1 and 3 show that D-DAWN-path is polynomial when the maximum number of slots D is lower than or equal to 2, and becomes NP-complete when $D \geq 3$. Theorem 4 proves that D-DAWN-request is NP-complete for $D = 2$.

Theorem 4. *The 2-DAWN-request decision problem is NP-complete*

Proof. Let $I_{SAT} = (U, W)$ be an instance of 3-SAT where $U = \{x_1, \dots, x_n\}$ denotes a set of variables and $W = \{c_1, \dots, c_m\}$ a set of clauses. For each variable $x_i \in U$ let $H_{x_i} = (X, Y, E)$ be the complete bipartite graph $K_{3,2}$ such that $X = \{(1, x_i), (1, \bar{x}_i)\}$ and $Y = \{(2, x_i), (2, \bar{x}_i)\}$. For each clause $c_i = \{l_1, l_2, l_3\} \in W$, let $F_{c_i} = (X, Y, E)$ be the complete bipartite graph $K_{3,2}$ such that $X = \{(1, c_i), (2, c_i)\}$ and $Y = \{(c_i, l_1), (c_i, l_2), (c_i, l_3)\}$. Note that l_1 to l_3 can be positive or negative literals, each literal corresponding to a variable $x_i \in U$. Let $I = (G, R)$ be a 2-DAWN-request instance with $V(G) = \{\{\bigcup_{x_i \in U} V(H_{x_i})\} \cup \{\bigcup_{c_i \in W} V(F_{c_i})\}\}$ and $E(G) = \{\{\bigcup_{x_i \in U} E(H_{x_i})\} \cup \{\bigcup_{c_i \in W} E(F_{c_i})\} \cup \{(c_i, l), (2, l)\}$

$\{c_i \in W, l \in c_i\}$. Figure 5 presents an example of graph G constructed from a 3-SAT instance $I_{SAT} = (U, W)$. The requests collection R contains exactly all the requests of the form $((1, c_i), (2, c_i)) | c_i \in W$, and $((1, l), (2, l))$ where l is a literal corresponding to a variable $x_i \in U$.

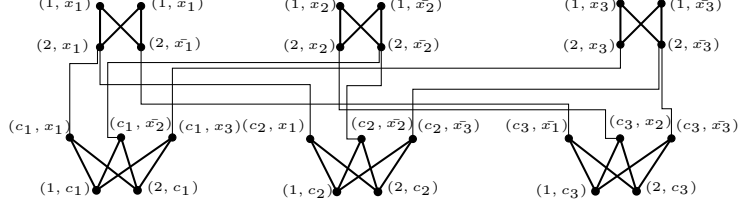


Fig. 5. graph constructed from $I_{SAT} = (U, W)$ with $U = \{x_1, x_2, x_3\}$ and $W = \{c_1, c_2, c_3\}$ with $c_1 = \{x_1, \bar{x}_2, x_3\}$, $c_2 = \{x_1, \bar{x}_2, \bar{x}_3\}$ and $c_3 = \{\bar{x}_1, x_2, \bar{x}_3\}$.

Let d be valid date assignment d for I using only 2 slots. We assign to any literal l the value "True" if and only if $(1, l)$ emits at slot 1, and "False" otherwise. Given a clause $c_i \in W$, exactly one node of the form (c_i, l) emits at slot 2. This node is adjacent to a node $(2, l)$, which could receive the message of the request $((1, l), (2, l))$ at slot 1 only. Thus $(1, l)$ is true and c is satisfied.

Reciprocally, suppose that I_{SAT} admits a solution. For each literal l fixed at "True", let us assign the date 1 to vertex $(1, l)$ and the date 2 to $(2, -l)$. Let c_i be a clause from W . Date 1 is assigned to vertex $(1, c_i)$. Date 2 must be assigned to exactly one adjacent vertex of $(1, c_i)$. We can choose any couple with the corresponding literal l fixed at "True". This date is available since (c_i, l) is only adjacent to $(2, c_i)$ (the destination) and $(2, l)$, which has already received the message at slot 1. Since 3-SAT is NP-complete and 2-DAWN-request is in NP, 2-DAWN-request is indeed a NP-complete problem. \square

Thus we have shown that knowledge of the routing policy plays a role in the complexity of both problems, since the limit between polynomiality and NP-completeness is located between 2 and 3 for DAWN-path, but between 1 and 2 for DAWN-request.

4 Solving instances with a bounded number of requests

We give a polynomial algorithm for min-DAWN-path problem and min-DAWN-request problem when the number of requests is bounded by above by a constant K . The following notation and definition will be used :

- For $i \in [1, n]$, let $\pi_i(t)$ denotes the i th element of a n -tuple $t = (x_1, x_2, \dots, x_n)$.
- The *contracted form* of a tuple (x_1, x_2, \dots, x_n) is the tuple $(x_i)_{(i \in [1, n]) \wedge (x_i \neq x_{i-1})}$.

We propose a polynomial algorithm to solve instances I with a number of requests bounded by K . We build a state graph, where each vertex describes a possible state of the network at a given slot. An edge links X and Y if and only if one can go from state X to state Y or reciprocally in only one slot. For a given min-DAWN-path instance $I = (G, R = (r_1, r_2, \dots, r_k), P)$ with $|R| \leq K$ the state graph $S(I)$ is defined as follows:

- the vertex set is the cartesian product $P(r_1) \times P(r_2) \cdots \times P(r_K)$. A vertex $X = (x_1, x_2, \dots, x_K)$ indicates that for each $i \leq K$, the message of request r_i has reached the node $\pi_i(X) = x_i \in V(G)$.
- there is an edge between $X = (x_1, x_2, \dots, x_K)$ and $Y = (y_1, y_2, \dots, y_K)$ of $S(I)$ if and only if the simultaneous emission of nodes $\{x_i | x_i \neq y_i, 1 \leq i \leq K\}$ allows to deliver each message from $x_i \neq y_i$ to y_i in one slot only. Formally, for $X = (x_1, x_2, \dots, x_K)$ and $Y = (y_1, y_2, \dots, y_K)$, $(X, Y) \in E(S(I))$ if we have, for each i such that $x_i \neq y_i$:
 - x_i and y_i are immediately consecutive in $P(r_i)$,
 - there is no $j \neq i$, such that $x_j \neq y_j$ and $\{x_j, y_i\} \in E(G)$,
 - for each $j \neq i$ such that $x_j \neq y_j$, we have $|x_i, y_i, x_j, y_j| = 4$.

The state graph $S(I)$ of a min-DAWN-request instance I is constructed according to the same method, except that the set of vertices is the set $V(G) \times V(G) \times \cdots \times V(G) = V(G)^K$. These state graphs can be constructed in a polynomial time, since K is a constant. We can distinguish two vertices in $S(I)$: the *source* (s_1, s_2, \dots, s_K) and the *sink* (t_1, t_2, \dots, t_K) where s_i and t_i are respectively the source and the target of the request r_i for any $i \in [1, K]$.

We conclude the section with this theorem:

Theorem 5. *min-DAWN-path and min-DAWN-request can be solved by a polynomial-time algorithm with complexity $O(n^{2K})$ when the number of requests is bounded by above by a constant K .*

Proof. (sketch of the proof) Consider $I = (G, R = (r_1, r_2, \dots, r_K), P)$ a min-DAWN-path instance with $|R| \leq K$, and let us construct the state graph $S(I)$.

One may check that a shortest path between the source and the sink in $S(I)$ can be associated with an optimal conflict-free date assignment and reciprocally. Such a path can be found with a $O(n^{2K})$ complexity algorithm. \square

5 Conclusion and perspectives

We have studied the complexity of the request satisfaction problem in a synchronous radio network. Table 1 summarises the results of this paper.

We have suggested other results [8] on particular cases i.e. on dynamic network, or when requests cannot be paused as soon as they have started. Possible perspective for this research work consist in studying the complexity of DAWN-path and DAWN-request on specific topologies, in order to discover polynomial cases even when the number of requests is unbounded. Particularly the complexity when the network is a chain is an open question (however for this case, we

DAWN-path:		Complexity:	DAWN-request:	
Min-DAWN-path		NP-hard (even in trees), not approximable within some constant factor.	Min-DAWN-request	
D-DAWN-path	$D \leq 2$	Polynomial	$D \leq 1$	D-DAWN-request
	$D \geq 3$	NP-complete	$D \geq 2$	
min-DAWN-path, $ R \leq K$		Polynomial : $O(n^{2K})$	min-DAWN-request, $ R \leq K$	

Table 1.

have a constant factor approximation algorithm). Moreover, finding heuristics with performance guarantee for difficult instances constitutes a natural extension of this work.

References

1. N. Alon, A. Bar-Noy, N. Linial, and D. Peleg. A lower bound for radio broadcast. *J. Comput. Syst. Sci.*, 43(2):290–298, 1991.
2. 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*, 2006.
3. 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)*, pages 103–106, May 2005.
4. G. Chelius. *Architectures et Communications dans les réseaux spontanés sans fil*. PhD thesis, INSA de Lyon, INRIA Rhône Alpes, France, April 2004.
5. I. Chlamtac and S. Kutten. On broadcasting in radio networks - problem analysis and protocol design. *IEEE Transactions on Communications*, 33:1240–1246, 1985.
6. I. Chlamtac and O. Weinstein. The wave expansion approach to broadcasting in multihop radio network. *IEEE Transaction Communication*, (39):426–433, 1991.
7. T. Cormen, C. Leiserson, R. Rivest, and C. Stein. *Introduction to Algorithms, 2nd edition*. Dunod, 2001.
8. B. Darties. *Problèmes algorithmiques et de complexité dans les réseaux sans fil*. PhD thesis, LIRMM, Université Montpellier 2, France, December 2007.
9. B. Darties and J. Palaysi. Satisfaction de requêtes par affectation de dates d'émissions dans les réseaux radios. In *Rencontres francophones du Parallélisme (RenPar'17)*, pages 157–163, 2006.
10. P. Fraigniaud and E. Lazard. Methods and problems of communication in usual networks. In *Proceedings of the international workshop on Broadcasting and gossiping*, pages 79–133. Elsevier North-Holland, Inc., 1994.
11. M. R. Garey and D. S. Johnson. *Computers and Intractability: A guide to the theory of NP-completeness*. Freeman, 1979.
12. S.M. Hedetniemi, S.T. Hedetniemi, and A.L. Liestman. A survey of gossiping and broadcasting in communication networks. *Networks*, 18:319–349, 1986.
13. D. R. Kowalski and A. Pelc. Centralized deterministic broadcasting in undirected multi-hop radio networks. In *APPROX-RANDOM*, pages 171–182, 2004.