

## Min-energy Broadcast in Kinetic *Ad hoc* Networks

J.M. Díaz-Báñez · R. Fabila-Monroy ·  
D. Flores-Peñaloza · M. A. Heredia · J.  
Urrutia

Received: date / Accepted: date

**Abstract** This paper studies the problem of computing an optimal range assignment in an *ad-hoc* wireless network for a kinetic scenario (KANET), where stations are modeled by points moving continuously in a  $d$ -dimensional space,  $d \geq 1$ . We assume that the motion of each object can be specified by a low-degree algebraic trajectory. We present results concerning the minimum energy consumption problem for a broadcast operation, where the cost of transmission is proportional to the  $\alpha$ -power of the Euclidean distance,  $\alpha \geq 1$ . An  $O(n^3 \log n)$ -time algorithm for this problem is presented, under the assumption that all stations have equally sized transmission ranges. However, we prove that the general version of the problem is NP-hard, and it is not approximable within a sub-logarithmic factor (unless  $P=NP$ ). We then consider a special case of the general problem (also NP-hard), and present a polynomial time approximation method whose approximation factor is  $(\ln n + 1)m$ , where  $m$  is the size of the optimum solution.

---

J.M. Díaz-Báñez  
Depto. de Matemática Aplicada II, Universidad de Sevilla, Sevilla, España  
E-mail: dbanez@us.es

R. Fabila-Monroy  
Universidad Nacional Autónoma de México, México D.F., México  
E-mail: ruy@ciencias.unam.mx

D. Flores-Peñaloza  
Universidad Nacional Autónoma de México, México D.F., México  
E-mail: dflores@matem.unam.mx

M. A. Heredia  
Universidad Nacional Autónoma de México, México D.F., México  
E-mail: marco@ciencias.unam.mx

J. Urrutia  
Instituto de Matemáticas, Universidad Nacional Autónoma de México, México D.F., México  
E-mail: urrutia@matem.unam.mx

**Keywords** Power assignment · Wireless *ad hoc* networks · Kinetic · Algorithms

## 1 Introduction

In recent years, optimization problems in ad-hoc wireless networks have attracted significant attention, in part due to their potential applications in civil and military domains ([Estrin et al(1999)Estrin, Govindan, Heidemann, and Kumar, Rabaey et al(2000)Rabaey, Ammer, da Silva, Patel, and Roundy]). Typically, the radio stations in such a network have a limited energy resource (battery for example), and consequently, energy efficiency is an important design consideration for these networks ([Clementi et al(2001)Clementi, Crescenzi, Penna, Rossi, and Vocca, Das et al(2006)Das, Das, and Nandy]). On the other hand, the rapidly expanding technology of cellular communications, wireless LAN, and satellite services, adds movement to the stations, making it necessary to extend the concept of *static* ad-hoc wireless networks to *mobile* ad-hoc wireless networks. The mobility of nodes (deterministic or random) adds another dimension in the complexity of the classic problems.

Briefly, a mobile ad hoc network (MANET) is an autonomous system of mobile routers and hosts connected by wireless links. Most of the papers about MANETs consider the network self-organizing and self-configuring, with nodes establishing the necessary routes among themselves without the help of a fixed infrastructure. A general MANET may show very heterogeneous spatial distribution of nodes, from locally very sparse to very dense; and very heterogeneous node mobility pattern, from low to high mobility systems. [Camp et al(2002)Camp, Boleng, and Davies.] gave an excellent survey on mobility models for ad hoc networks.

However, in some applications it is not unrealistic to assume that devices do not change their trajectories for communication purposes, like in [Zhao et al(2004)Zhao, Ammar, and Zegura]. A model to abstract this kind of problems is needed.

Broadcasting is a common communication mechanism. A *broadcast communication* is a task initiated by a source station which has to disseminate a message to all stations in the network ([Peleg(2007)]). Broadcast data have typically a temporal and spatial relevance. In this paper, we consider only the temporal relevance of data, and assume that information becomes irrelevant after a certain period of time, i.e. its *lifetime* ([Khelil et al(2007)Khelil, Marrón, Becker, and Rothermel]).

Motivated by the previously exposed ideas, we consider an abstraction for source-initiated broadcasting of messages in a specific wireless network, named *kinetic ad-hoc wireless networks* (KANETs), for which the stations are moving according to fixed trajectories. Such abstraction can be applied to real life in, for example, satellite networks ([Resende and Pardalos(2006)]), where each mobile object in the set has a published flight plan and the broadcasting time is bounded by a time interval or lifetime. We believe that the study of this model is interesting from both theoretical and practical points of view.

For simplicity and computational purposes, we restrict the movement of the stations to rectilinear trajectories on the plane and constant velocity. However, the same results hold for scenarios where the speeds are finite and the motion is described by some low-degree algebraic curve, assuming only that, for any two objects, we can efficiently determine when they meet or separate for the first time.

We extend the static model of [Peleg(2007)] to a simplified kinetic system model as follows: Let  $S = \{s_1, s_2, \dots, s_n\}$  be a set of  $n$  moving points on the plane, which model our radio stations, moving independently, continuously, and with constant speed on a straight line each one. For a mobile station  $s_i \in S$ , the *transmission range* of  $s_i$ ,  $C_{s_i}$  is a circle of radius  $r \geq 0$  centered at  $s_i$ . A station  $s_j$  can receive a transmission from  $s_i$  at time  $t$  if and only if  $s_j \in C_{s_i}$  at time  $t$ . A *transmission range assignment* for  $S$  is a function  $R : S \rightarrow \mathbb{R}^+$  such that the station  $s_i$  has assigned a range size of  $R(s_i)$ .

The following conditions about message transmission are assumed throughout this paper: a message transmission can be completed in an instant of time; if  $s_i$  receives or generates a message at time  $t$ , then it will pass the message to every station with whom it can communicate at any time  $t' \geq t$ .

The power (energy) needed to correctly transmit data from a station  $s$  to a station  $t$  depends on the term  $d(s, t)^\alpha$ , where  $d(s, t)$  is the Euclidean distance between  $s$  and  $t$ , and  $\alpha \geq 1$  is the *distance-power gradient*. In an ideal environment  $\alpha = 2$ , but depending on the environment conditions may exceed the value of 6 ([Pahlavan and Levesque(2005)]).

Given a source station  $s \in S$ , and a message  $M$  generated in  $s$  at time  $t_0$ , we are interested in completing a broadcast operation of  $M$  to the rest of the stations. This paper studies the following problems:

- (1) *connectivity problem*: Given a transmission range assignment  $R$  for  $S$ , decide if  $M$  arrives to every station in  $S$ .
- (2) *min-equal-range problem*: Find the minimum value  $r$  needed to perform the broadcast, supposing that each station has a range radius of  $r$ .
- (3) *min-sum problem*: Find a transmission range assignment  $R$ , such that the broadcast is performed and the value  $\sum_{s_i \in S} R(s_i)^\alpha$  is minimized,  $\alpha \geq 1$ .
- (4) *min-sum-binary problem*: Compute the minimum number of stations needed to perform the broadcast, assuming that each station can only transmit with range radii either 1 or 0.

The first three problems have been widely studied for the static case, and the objective functions presented in [Clementi et al(2001)] Clementi, Crescenzi, Penna, Rossi, and Vocca] or [Wang and Li(2006)] have been taken into account. In general, in the static case, a directed transmission graph is directly obtained from the transmission range assignment  $R$ , then a specific connectivity property of that graph is searched. However, these strategies cannot be generalized to the kinetic case, as the transmission of messages depends also on the time of transmission.

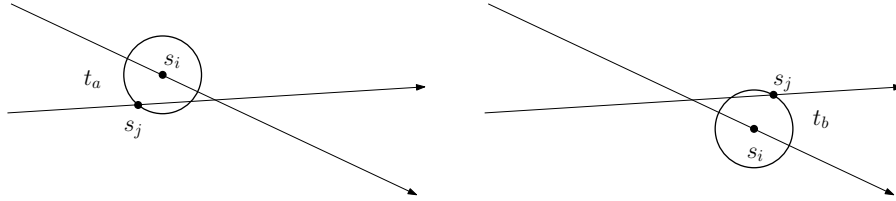
This paper is organized as follows: Section 2 presents an algorithm for solving the connectivity problem. Section 3 proposes an  $O(n^3 \log n)$ -time algorithm to solve the min-equal-range problem. In Section 4 we prove that min-sum is an

NP-hard problem, and discuss the impossibility of finding an algorithm whose approximation ratio is a sub-logarithmic factor from an optimum solution, unless  $P=NP$ . Next, in Section 5 we move to the min-sum-binary problem, showing that min-sum-binary is also NP-hard, and then propose an approximation algorithm that achieves an approximation factor of  $(\ln n + 1)m$ , where  $m$  is the size of the optimum solution. Finally, we present our conclusions in Section 6.

## 2 Connectivity

To solve the connectivity problem we propose an algorithm based on Dijkstra's algorithm, which, as a side result, also computes the first time at which each station receives the message.

Given a transmission range assignment  $R$ , since two different stations  $s_i$  and  $s_j$  move along different lines,  $s_j$  can lie in  $C_{s_i}$  only during one time interval, at which  $s_i$  can send a message to  $s_j$ . The *transmission interval* of  $s_j$  from  $s_i$ ,  $I_{s_i}(s_j)$ , is the time interval  $[t_a, t_b]$ , where  $t_a \in \mathbb{R}^+$  is the first instant of time in which  $s_j$  lies in  $C_{s_i}$ , and  $t_b \in \mathbb{R}^+$  is the instant of time in which  $s_j$  leaves  $C_{s_i}$ . See Fig. 1.



**Fig. 1** Times defining  $I_{s_i}(s_j) = [t_a, t_b]$ .

Observe that  $I_{s_i}(s_j)$  can be an empty interval and one of the following containments is always true  $I_{s_i}(s_j) \subseteq I_{s_j}(s_i)$  or  $I_{s_j}(s_i) \subseteq I_{s_i}(s_j)$ . Both are true if and only if  $s_i$  and  $s_j$  have equally sized transmission ranges, or both  $I_{s_i}(s_j)$  and  $I_{s_j}(s_i)$  are empty.

We suppose that the trajectory of each  $s_i \in S$  is given to us in a way that computing the *transmission interval* of two stations can be done efficiently, consequently this operation is assumed to be computed in constant time.

The *connectivity graph*  $G_R$  generated by a set of mobile stations  $S$  and a transmission range assignment  $R$ , is the directed graph having  $S$  as vertex set, and there is an arc from  $s_i$  to  $s_j$  in  $G_R$  if and only if  $I_{s_i}(s_j)$  is a non empty interval. This arc is labeled with the interval  $I_{s_i}(s_j)$ . See the left side of Fig. 2 for an example.

To solve the connectivity problem it is enough to show that  $G_R$  has a directed spanning tree rooted at  $s$ , with suitable labels on the arcs.

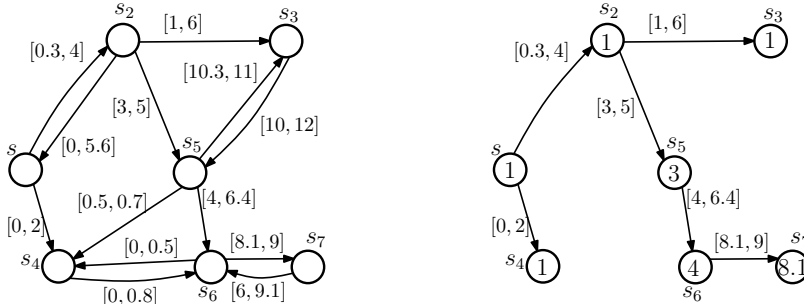
If  $t_i$  is the first time in which the station  $s_i$  receives the message  $M$ , then  $s_i$  can pass the message to a station  $s_j$  if  $t_i \leq t_b$ , where  $[t_a, t_b] = I_{s_i}(s_j)$ . This

concept can be expressed in  $G_R$  in the following way: assign the value  $t_i$  to the vertex  $s_i$ ; consider the arc from  $s_i$  to  $s_j$ , labeled  $[t_a, t_b]$ , and assign the time  $t_j$  to  $s_j$  (the time at which  $s_j$  first receives  $M$  from  $s_i$ ), where  $t_j = t_i$  if  $t_i \in [t_a, t_b]$  or  $t_j = t_a$  if  $t_i \leq t_a$ .

The following algorithm, which we call *IS\_CONNECTED*, solves the connectivity problem: Construct the connectivity graph  $G_R$  of  $S$ ; then assign the time  $t_0$  (time at which the source generates the message  $M$ ) to the vertex  $s$ , the value of  $\infty$  to the other vertices and run a modified Dijkstra's algorithm ([Cormen et al(2001)Cormen, Leiserson, and Rivest]) from  $s$ , sorting the vertices in the priority queue (of the Dijkstra's algorithm) by the time at which they receive  $M$ .

Typical Dijkstra's algorithm assures that the vertex going out of the priority queue has assigned the minimum distance to the source. Thus, *IS\_CONNECTED* assures that each vertex going out of the queue has assigned the minimum time at which it receives the message  $M$ . Therefore, if the tree obtained from *IS\_CONNECTED* is a spanning tree of  $G_R$ , then the broadcast from  $s$  will succeed (see right side of Fig. 2). In this way, we transform the connectivity problem to a shortest path like problem. As a side result, we obtain the first time at which each vertex receives the message. The correctness and complexity of *IS\_CONNECTED* follow from the correctness and complexity of Dijkstra's algorithm. As  $G_R$  could be a complete graph, then the total running time of *IS\_CONNECTED* is  $O(n^2)$  and we arrive to the following result:

**Theorem 1** *The connectivity problem can be solved in  $O(n^2)$  time.*



**Fig. 2** The graph  $G_R$  of a kinetic network and the spanning tree obtained by the algorithm ( $t_0 = 1$ ).

### 3 Equally sized ranges

In this section we describe an  $O(n^3 \log n)$  algorithm to solve the min-equal-range problem. We recall that the problem consist of finding the minimum

value  $r$  needed to perform a broadcast, supposing that each station has a range radius of  $r$ .

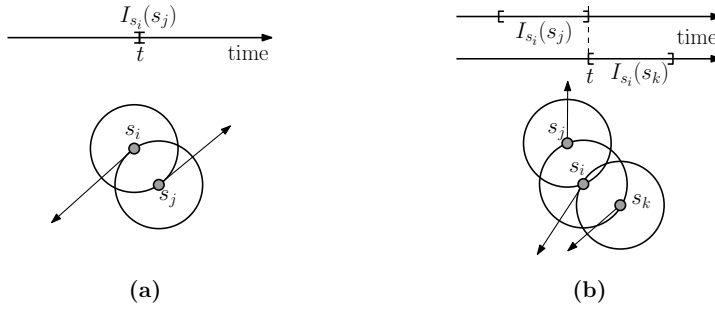
As the radius of  $C_{s_i}$  is equal to the radius of  $C_{s_j}$ , then  $I_{s_i}(s_j) = I_{s_j}(s_i)$ , so we will use  $I_{s_i}(s_j)$  and  $I_{s_j}(s_i)$  interchangeably. This fact will transform the connectivity graph into an undirected graph.

For any given  $r \geq 0$ , we call  $G_r$  the connectivity graph of  $S$  obtained by assigning to each station the transmission radius  $r$ , and  $T_r$  the tree obtained by running the algorithm *IS-CONNECTED* on  $G_r$ . The problem is then reduced to finding the minimum radius  $r_{MIN}$  for which  $T_{r_{MIN}}$  is a spanning tree of  $G_{r_{MIN}}$ .

The key idea of our algorithm is to calculate a discrete set of possible values for  $r_{MIN}$ , and then search over those values. We focus on all radii  $r$  such that  $T_r$  and  $T_{r-\epsilon}$  could be different, for all small  $\epsilon > 0$ , implying that  $T_{r-\epsilon}$  might not be a spanning tree of  $G_{r-\epsilon}$ . We call these radii *critical radii*.

We say that  $r$  is a critical radius for  $S$  if by assigning  $r$  to all stations in  $S$ , one of the following cases arises:

- Two stations in  $S$ ,  $s_i$  and  $s_j$ , have only one instant  $t$  of connection ( $I_{s_i}(s_j) = [t, t]$ ). See Fig. 3a.
- Three different stations in  $S$ ,  $s_i$ ,  $s_j$  and  $s_k$ , have the property of  $I_{s_i}(s_j) \cap I_{s_i}(s_k) = [t, t]$ . See Fig. 3b.



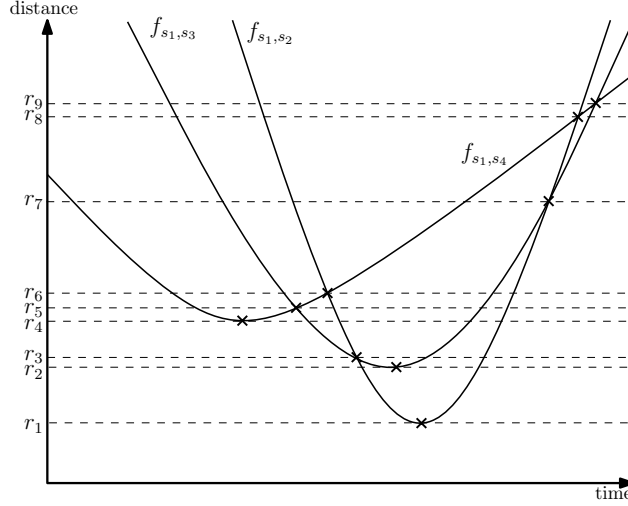
**Fig. 3** Example of the two cases of critical radius.

A radius of type a) corresponds to the addition of an edge in  $G_r$  that is not present in  $G_{r'}$ , with  $r' < r$ . The type b) corresponds to radii where an edge of  $T_r$  is possibly not present in  $T_{r'}$ , with  $r' < r$ . The set of all critical radii of  $S$  is denoted by  $CR(S)$ .

Given two different stations,  $s_i$  and  $s_j$ , let  $f_{s_i, s_j}(t)$  be the squared Euclidean distance between  $s_i$  and  $s_j$  at time  $t$ . As  $s_i$  and  $s_j$  move along lines,  $f_{s_i, s_j}$  is a quadratic polynomial in  $t$ . Observe that  $f_{s_i, s_j} = f_{s_j, s_i}$ .

Consider the arrangement of the  $n - 1$  functions involving  $s_i$  ( $\{f_{s_i, s_j} \mid i \neq j\}$ ). Any two of these functions intersect at most twice, then the arrangement contains  $O(n^2)$  intersections. Each of the  $O(n^2)$  intersections gives us a (squared) radius that corresponds to the type b) of critical radii, and each of

the  $n - 1$  function minima gives us a (squared) radius that corresponds to the type a) of critical radii. Refer to Fig. 4. Since we have  $n$  different arrangements, then the size of  $CR(S)$  is  $O(n^3)$ . Assuming that we can obtain the minima of any of these functions, and calculate the intersection of two of these functions in constant time; then we can compute  $CR(S)$  in  $O(n^3)$  time.



**Fig. 4** Example of critical radii corresponding to  $s_1$ .

The algorithm *MIN\_RADIUS*, for solving the min-equal-range problem, can be defined as follows: Compute the  $CR(S)$  set ( $O(n^3)$  time), sort the elements in  $CR(S)$  ( $O(n^3 \log n)$ ), and look for the minimum radius  $r_{MIN}$  in which  $T_{r_{MIN}}$  is a spanning tree of  $G_{r_{MIN}}$ ; by using binary search and applying the *IS\_CONNECTED* algorithm ( $O(n^2)$ ) at each step. The total running time is then  $O(n^3 \log n)$ .

In summary, we have proven the following result:

**Theorem 2** *The min-equal-range problem can be solved in  $O(n^3 \log n)$  time.*

#### 4 The general min-sum problem

This section focus on showing that the general version of min-sum problem is intractable (unless  $P=NP$ ), by reducing the well-known weighted-set-cover problem ([Chvatal(1979), Cormen et al(2001)Cormen, Leiserson, and Rivest]) to the min-sum problem.

From the results on [Clementi et al(2001)Clementi, Crescenzi, Penna, Rossi, and Vocca], about the static version of min-sum, the NP-harness of min-sum can be implied for distance power gradient  $\alpha > 1$  and stations lying on a space of dimension  $d \geq 2$ . However, unlike the static case, we can prove that the min-sum problem

on a kinetic scenario is not solvable in polynomial time (unless  $P=NP$ ), even if  $\alpha = 1$ . In the same way, although the static problem is approximable within a constant factor on the plane, for  $\alpha \geq 2$ , we show that the kinetic problem is not approximable within a sublogarithmic factor.

An instance of the weighted-set-cover problem consists of a finite set  $A$ ; a family  $B$  of subsets of  $A$ , such that every element of  $A$  belongs to at least one subset in  $B$ ; and a weight function  $w : B \rightarrow \mathbb{R}^+$ . We say that a subset  $c \in B$  *covers* its elements with weight  $w(c)$ . The problem is to find a minimum-weight *cover*  $\mathcal{C} \subseteq B$  whose members cover all of  $A$ , where the weight of  $\mathcal{C}$  is  $\sum_{c \in \mathcal{C}} w(c)$ .

It is known that the decision version of weighted-set-cover problem is NP-complete and the optimization (minimization) version is NP-hard ([Chvatal(1979), Cormen et al(2001)Cormen, Leiserson, and Rivest]). We can prove the NP-hardness of the min-sum problem by reducing the weighted-set-cover problem to the min-sum problem.

For this, take an instance of the weighted-set-cover problem:

$A = \{a_1, a_2, \dots, a_k\}$  (the covered set),  $B = \{b_1, b_2, \dots, b_l\}$  (the covering set) and the weight function  $w : B \rightarrow \mathbb{R}^+$ . The transformation to an instance of the min-sum problem is as follows (see Figure 5):

Take a set of stations  $SB = \{sb_1, sb_2, \dots, sb_l\}$ , collinear in a horizontal line  $\mathcal{L}$ , at distance  $\delta$  (large enough) from each other, and moving leftwards with the same speed. We associate each station  $sb_i$  of  $SB$  to the element  $b_i$  in  $B$ . The source  $s$ , that generates the message  $M$  at time  $t_0$ , moves rightwards in a line parallel to  $\mathcal{L}$  whose distance to  $\mathcal{L}$  is 1, in a way that after a certain time, say  $t_1 > t_0$ ,  $s$  could have transmitted the message  $M$  to each station of  $SB$  if its transmission radius were equal to 1.

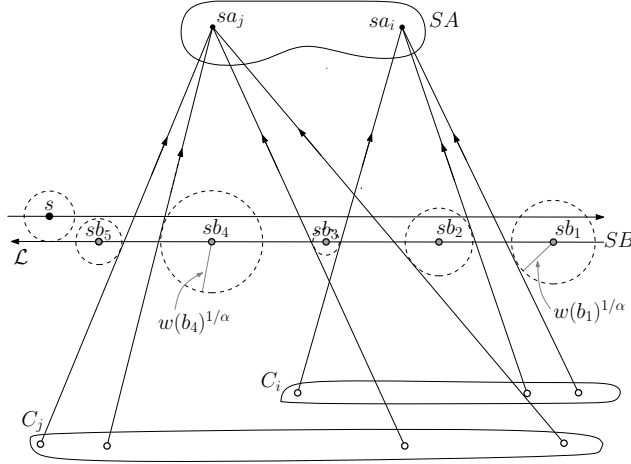
Now let  $SA = \{sa_1, sa_2, \dots, sa_k\}$  be a set of static points in the plane far away from the trajectories of  $SB$  and  $s$ , that we call the *meeting points*. Each element  $a_i$  in  $A$  is associated to the point  $sa_i$  of  $SA$ .

Finally we create the *cover relation* by adding station sets  $C = C_1 \cup C_2 \cup \dots \cup C_k$  in the following way: Take an element  $a_i$  in  $A$  and take  $B_i$  as the set of elements in  $B$  that covers  $a_i$ . For each  $b_l$  in  $B_i$  we create a new station  $c_{i,l}$  in the set  $C_i$ , in a way that:  $c_{i,l}$  passes at distance  $w(b_l)^{1/\alpha}$ ,  $\alpha \geq 1$ , from  $sb_l$  at time  $t_{i,l} > t_1$ , passes over  $sa_i$  at time  $t_{a_i} > t_{i,l}$ , and at any time the minimum distance between  $sb_l$  and  $c_{i,l}$  is  $w(b_l)^{1/\alpha}$ . Intuitively, every station  $sb_l$  that *covers*  $sa_i$  sends a station to  $sa_i$ , and all stations in  $C_i$  arrive to  $sa_i$  at the same time ( $t_{a_i}$ ). Notice that we can take suitable times and distances so all the events occur independent from each other.

In some sense, we only permit that each element  $sb_l \in SB$  have two choices, transmit with radius  $w(b_l)^{1/\alpha}$ , or 0. On the other hand, all the stations in  $C_i$  can receive  $M$  at time  $t_{a_i}$  if one of them has it and transmit with radius 0.

Clearly, the above transformation can be done in polynomial time. A range assignment  $R$  that minimizes  $\sum_{s_i \in S} R(s_i)^\alpha$  and allows a broadcast from  $s$ , is one that: assign radius 1 to  $s$ ; takes  $\mathcal{D} \subseteq SB$  such that  $\sum_{sb_l \in \mathcal{D}} R(sb_l)^\alpha$  is minimized, where  $sb_l \in \mathcal{D}$  has assigned a radius  $R(sb_l) = w(b_l)^{1/\alpha}$ ; leaves the stations of  $SB \setminus \mathcal{D}$  with radius 0; and assures that at least one of the elements





**Fig. 5** Reduction of weighted-set-cover problem to min-sum problem.

in each  $C_i$  have the message  $M$ . But this assignment maps to a subset of  $B$  of minimum weight that covers  $A$ , solving then the weighted-set-cover problem. Since weighted-set-cover problem can only be approximated to within a  $\ln |A| + 1$  factor (unless  $P=NP$ ) ([Feige(1998)]), no polynomial time approximation algorithm for min-sum problem achieves a smaller approximation factor. The following result can then be established:

**Theorem 3** *The min-sum problem is NP-hard and, unless  $P=NP$ , it is not approximable within a sub-logarithmic factor.*

## 5 The min-sum-binary problem

In this section we first prove that the min-sum-binary problem also belongs to the NP-hard complexity class, and then we describe an approximation approach based on the well-known greedy algorithm for the weighted-set-cover problem ([Chvatal(1979), Cormen et al(2001)Cormen, Leiserson, and Rivest]).

### 5.1 NP-hardness

A similar proof to that of the NP-hardness of min-sum, shows that the min-sum-binary problem is NP-hard. By taking an instance of the weighted-set-cover problem, but with all the weights of the elements in  $B$  set to 1, we obtain an instance of the classic set-cover problem ([Garey and Johnson(1979)]). In the set-cover problem we need to find a minimum-size subset of elements of  $B$  whose members cover all of  $A$ . The set-cover problem is also NP-hard and approximable within a  $\ln |A| + 1$  factor (unless  $P=NP$ ) ([Garey and Johnson(1979), Cormen et al(2001)Cormen, Leiserson, and Rivest, Feige(1998)]).

It is straightforward to see that the construction in the previous section can be used to reduce the set-cover problem to the min-sum-binary problem, by assigning radius 1 or 0 to the elements of  $SB$ .

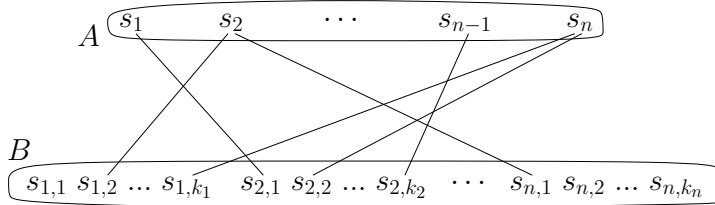
## 5.2 Approximation algorithm

As we cannot obtain a polynomial time approximation algorithm (for the min-sum-binary problem) that achieves a sub-logarithmic approximation ratio, then we propose an approach based on a greedy algorithm for the weighted-set-cover problem ([Chvatal(1979), Cormen et al(2001)Cormen, Leiserson, and Rivest]) whose approximation ratio is  $\ln |A| + 1$ .

Roughly speaking, at each step of the greedy algorithm, the subset that covers the most elements with the lowest weight is chosen. Specifically, for every subset, the number of elements it covers (not yet covered) is divided by its weight. The subset with the highest ratio is then chosen.

In order for  $s_i$  to be able to transmit the message  $M$  to another station  $s_j$ ,  $s_i$  must have received  $M$  before the transmission interval between them ends. Then for each station  $s_i$  we consider the times  $t_{i,1} \leq t_{i,2} \leq \dots \leq t_{i,k_i}$  at which  $s_i$  loses connectivity with a station (with  $k_i \leq n-1$ , as we only consider those stations that are ever within transmission range of  $s_i$ ). We denote by  $s_{i,j}$ , the set of stations that lose connectivity with  $s_i$  at or after time  $t_{i,j}$ . By definition,  $s_{i,j+1} \subset s_{i,j}$ .

We construct an instance of the weighted-set-cover problem. The set  $A = S$  of all stations is the set to be covered. As the family  $B$  of subsets used to cover  $S$ , we use the subsets  $s_{i,j}$ , as illustrated in Figure 6. We define the weight function  $w : B \rightarrow \mathbb{R}^+$  such that  $w(s_{i,j})$  is the minimum number of stations, including  $s_i$ , that must be turned on (have transmission range equal to 1) in order for  $s_i$  to receive the message at or before time  $t_{i,j}$ . We elaborate on this later.



**Fig. 6** Transforming min-sum-binary problem to weighted-set-cover problem (edges represent the cover relation).

Using the greedy algorithm for weighted set cover, at each step we choose the subset  $s_{i,j}$  with the best ratio between the number of new covered stations and  $w(s_{i,j})$ . Denote by  $\mathcal{D}'$  the set of stations (and times)  $s_{i,j}$  chosen by the greedy algorithm. It is known that the greedy solution is at most at a  $\ln |A| + 1$

factor of the optimal solution of the weighted set cover problem. In particular, it is at most at a  $\ln |A| + 1$  factor from any other solution.

Consider an optimal solution  $\mathcal{D}_O$  for the min-sum-binary problem. Let  $s_i \in \mathcal{D}_O$ . Assume  $s_i$  receives the message for the first time at time  $t$ . Replace  $s_i$  with  $s_{i,j}$ , where  $t_{i,j}$  is the first moment in time after  $t$  at which  $s_i$  loses connectivity with another station. Doing this for every element  $s_i \in \mathcal{D}_O$ , we obtain a covering set  $\mathcal{D}_O'$  for our weighted instance of set cover. Therefore:

$$(\ln |A| + 1) \left( \sum_{s_{i,j} \in \mathcal{D}_O'} w(s_{i,j}) \right) \geq \sum_{s_{i,j} \in \mathcal{D}_O'} w(s_{i,j}) \quad (1)$$

We then obtain a solution for the min-sum-binary problem from the set  $\mathcal{D}'$  as follows. For every  $s_{i,j}$ , we turn on the corresponding  $s_i$  as well as the other  $w(s_{i,j}) - 1$  stations that enable  $s_i$  to receive the message at or before time  $t_{i,j}$ . Denote by  $\mathcal{D}$  the set of stations turned on in this manner.

By construction of  $\mathcal{D}$ , we have that  $|\mathcal{D}| \leq \sum_{s_{i,j} \in \mathcal{D}'} w(s_{i,j})$ . Also for every  $s_{i,j} \in \mathcal{D}_O'$ , its cost cannot be greater than  $|\mathcal{D}_O|$ . Therefore  $\sum_{s_{i,j} \in \mathcal{D}_O'} w(s_{i,j}) \leq |\mathcal{D}_O| |\mathcal{D}_O|$  and together with (1), we have:

$$((\ln |A| + 1) |\mathcal{D}_O|) |\mathcal{D}_O| \geq |\mathcal{D}| \quad (2)$$

As  $|A| = n$ , our greedy algorithm thus yields an approximation ratio of  $(\ln n + 1)m$ , where  $m$  is the size of an optimal solution for the min-sum-binary problem.

We now describe how to obtain the weights  $w(s_{i,j})$ . For the time being, assume that all stations are turned on. For any moment in time  $t$ , let  $G_t$  be the graph with  $S$  as vertex set, with two stations adjacent if they are currently within the transmission range of each other.

Let  $t_0$  be the moment in time at which the source receives the message for the first time. Let  $t_1 \leq t_2 \leq \dots \leq t_p$  be the moments in time at which communication is gained or lost between any two stations.

Consider the sequence of graphs  $G_{t_0}, G_{t_1}, \dots, G_{t_p}$ . Two consecutive graphs in the sequence differ by an edge, that is either removed or added. Also since two stations meet each other at most once, the sequence has at most  $O(n^2)$  terms (for rectilinear trajectories, two stations meet each other at most once, then the sequence will have at most  $O(n^2)$  terms; for more complex trajectories the size of the sequence could increase).

Notice that the minimum number of stations that need to be turned on, in order for  $s_i$  to receive the message at time  $t_i$  or before, is none other than the minimum number of times that the message has been transmitted to reach  $s_i$  before or at time  $t_i$ .

Starting from the source in  $G_{t_0}$ , we then apply Dijkstra's algorithm and compute the distance of every station to the source. This distance (plus one) is the number of hops that the message must have made to reach a given station. For each station we keep track of: the number of hops, the path of stations and the time at which this was achieved.

We move from graph to graph in sequence. When we move to the next graph  $G_{t_l}$  and a new edge is added, a station may be able to receive the message in fewer hops. Suppose that the added edge was  $\{s_i, s_j\}$  and that, say  $s_i$ , received the message in  $h$  hops, where  $s_j$  received the message in more than  $h + 1$  hops. The station  $s_j$  can now receive the message from  $s_i$  in fewer hops. Also, any station currently adjacent to  $s_j$  might receive the message in fewer hops and so on. We compute these new values (keeping the old ones for further reference) by applying again Dijkstra's algorithm on the new graph  $G_{t_l}$ , with  $s_i$  as the starting vertex; with the exception that we relax only the edges that leave vertices that now receive the message in fewer hops. The distance of a station to  $s_i$  plus the number of hops used to reach  $s_i$  is the number of hops that can now be used to reach the station. This number may or may not be less than the current number of hops. If it is less, we add a new entry with the number of hops, time at which this was achieved and the path used. If linked lists are used to represent paths, this can be done in constant time per station. If the next graph comes from the deletion of an edge, we delete this edge and move on to the next graph.

Once we have visited all the graphs in the sequence, every vertex contains a decreasing list of number of hops, time and path. Using this list we can calculate the costs of each  $s_{i,j}$  at a linear cost per station, thus spending  $O(n^2)$  time in total.

Finally, since we have an instance of the weighted-set-cover problem with a set of  $n$  elements and a family of  $O(n^2)$  subsets, the greedy algorithm can be implemented to run in time  $O(n^4)$ . Thus the approximation algorithm runs in  $O(n^4)$  time in total. We end this discussion by stating the following theorem:

**Theorem 4** *Given a set of  $n$  kinetic stations, a  $(\ln n + 1)m$ -approximation of the min-sum-binary problem can be computed in  $O(n^4)$  time (for rectilinear trajectories), where  $m$  is the size of the optimum solution.*

## 6 Conclusions

In this paper, we worked on the problem of energy-efficient source-initiated broadcasting in a kinetic wireless network, where every station moves at constant velocity along a linear trajectory. First, we showed that the problem of minimizing the sum of the squares of the range radii for a broadcast operation is polynomial, when all stations have the same radius, and it is NP-hard in the general setting. We then presented a polynomial-time approximation algorithm for a special case of the min-sum problem for which the stations transmit with radius either 1 or 0. Our approach performs approximation within a factor  $(\ln n + 1)m$  of the optimal, where  $m$  is the size of the optimum solution.

It is worth mentioning that we never use the fact that the stations move at the same velocity, consequently, our results follow even when the speeds of the stations are different, as long as we forbid infinite speeds. On the other hand, the complexity of the problem does not change if we assume algebraic

motions instead of rectilinear motions, as long as intersections and calculation of distance between two trajectories can be computed efficiently.

The difference in complexity between the static and the kinetic setup, is expressed in the fact that the NP-hardness is maintained, even with the distance-power gradient  $\alpha = 1$ , and that the minimum approximation factor grows from constant to logarithmic, even with dimension  $d = 2$  ([Clementi et al(2001)Clementi, Crescenzi, Penna, Rossi, and Vocca]).

Finally, for future work, we would like to improve the approximation factor of Theorem 4. However, we believe that very little can be accomplished by approximation algorithms for the min-sum problem. In fact, we conjecture that it is not approximable within an even bigger factor than the logarithmic, showed in Theorem 3. On the other hand, we must note that the *IS\_CONNECTED* algorithm can be used to obtain a simple heuristic method for the min-sum problem, by applying a tabu search strategy ([Glover and Laguna(1997)]).

**Acknowledgements** This work was supported in part by Spanish Government under project MEC-MTM2006-03909.

## References

- [Camp et al(2002)Camp, Boleng, and Davies.] Camp T, Boleng J, Davies V (2002) A survey of mobility models for ad hoc network research. *Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications* 2(5):483–502
- [Chvatal(1979)] Chvatal V (1979) A greedy heuristic for the set covering problem. *Mathematics of Operations Research* 4(3):233–235
- [Clementi et al(2001)Clementi, Crescenzi, Penna, Rossi, and Vocca] Clementi A, Crescenzi P, Penna P, Rossi G, Vocca P (2001) On the complexity of computing minimum energy consumption broadcast subgraphs. In: 18th Annual Symposium on Theoretical Aspects of Computer Science, LNCS 2001, pp 121–131
- [Cormen et al(2001)Cormen, Leiserson, and Rivest] Cormen TH, Leiserson CE, Rivest RL (2001) *Introduction to Algorithms*. MIT Press and McGraw-Hill
- [Das et al(2006)Das, Das, and Nandy] Das GK, Das S, Nandy SC (2006) Range assignment for energy efficient broadcasting in linear radio networks. *Theoretical Computer Science* 352(1):332–341
- [Estrin et al(1999)Estrin, Govindan, Heidemann, and Kumar] Estrin D, Govindan R, Heidemann J, Kumar S (1999) Next century challenges: scalable coordination in sensor networks. In: *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pp 263 – 270
- [Feige(1998)] Feige U (1998) A threshold of  $\ln n$  for approximating set cover. *Journal of the ACM (JACM)* 45(4):634 – 652
- [Garey and Johnson(1979)] Garey MR, Johnson DS (1979) *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co.
- [Glover and Laguna(1997)] Glover F, Laguna M (1997) *Tabu Search*. Kluwer Academic Publishers
- [Khelil et al(2007)Khelil, Marrón, Becker, and Rothermel] Khelil A, Marrón P, Becker C, Rothermel K (2007) Hypergossiping: A generalized broadcast strategy for mobile ad hoc networks. *Ad Hoc Networks* 5(5):531–546
- [Pahlavan and Levesque(2005)] Pahlavan K, Levesque AH (eds) (2005) *Wireless information networks*. Wiley-Interscience
- [Peleg(2007)] Peleg D (2007) Time-efficient broadcasting in radio networks: A review. In: *Proceedings of the 4th International Conference on Distributed Computing and Internet Technology (ICDCIT)*, LNCS 4882, pp 1–18

- 
- [Rabaey et al(2000)Rabaey, Ammer, da Silva, Patel, and Roundy] Rabaey JM, Ammer MJ, da Silva JL, Patel D, Roundy S (2000) Picoradio supports ad hoc ultra-low power wireless networking. *Computer* 33(7):42–48
- [Resende and Pardalos(2006)] Resende MG, Pardalos PM (eds) (2006) *Handbook of Optimization in Telecommunications*. Springer
- [Wang and Li(2006)] Wang Y, Li X-Y (2006) Minimum power assignment in wireless ad hoc networks with spanner property. *Journal of Combinatorial Optimization* 11(1):99–112
- [Zhao et al(2004)Zhao, Ammar, and Zegura] Zhao W, Ammar M, Zegura E (2004) A message ferrying approach for data delivery in sparse mobile ad hoc networks. In: *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing (MOBIHOC)*, Tokyo, Japan, pp 187–198