Maximizing the Number of Broadcast Operations in Random Geometric Ad-Hoc Wireless Networks *

Tiziana Calamoneri† Andrea E. F. Clementi‡ Emanuele G. Fusco†
Riccardo Silvestri†

April 20, 2009

Abstract

We consider static ad-hoc wireless networks whose nodes, equipped with the same initial battery charge, may dynamically change their transmission range. When a node v transmits with range $r(v)$, its battery charge is decreased by $\beta\, r(v)^2$ where $\beta > 0$ is a fixed constant. The goal is to provide a range assignment schedule that maximizes the number of broadcast operations from a given source (this number is denoted as the *length* of the schedule). This maximization problem, denoted as MAX LIFETIME, is known to be NP-hard and the best algorithm yields worst-case approximation ratio $\Theta(\log n)$, where n is the number of nodes of the network.

We consider *random geometric* instances formed by selecting n points independently and uniformly at random from a square of side length $\sqrt{n}$ in the Euclidean plane.

We present an efficient algorithm that constructs a range assignment schedule having length not smaller than 1/12 of the optimum with high probability.

Then we design an efficient distributed version of the above algorithm where nodes initially know n and their own position only. The resulting schedule guarantees the same approximation ratio achieved by the centralized version thus obtaining the first distributed algorithm having *provably-good* performance for this problem.

1 Introduction

In static ad-hoc wireless networks, nodes have the ability to vary their transmission ranges (and, thus, their energy consumption) in order to provide good network connectivity and low energy consumption at the same time. More precisely, the transmission ranges determine a (directed) communication graph over the set V of nodes. Indeed, a node v, with range r, can transmit to another node w if and only if w belongs to the *disk* of radius r centered in v. The transmission range of a node depends, in turn, on the energy power supplied to the node. In particular, the power P_v required by a node v to correctly transmit data to another station w must satisfy the inequality (see [24]):

$$\frac{P_v}{\mathtt{dist}(v,w)^2} \geq \eta$$

*An extended abstract of this work has been presented at OPODIS 2007. Research partially supported by the European Union under the Project IP-FP6-015964 AEOLUS.

†Dipartimento di Informatica, Università degli Studi di Roma "La Sapienza". `E-mail: {calamo, fusco, silver}@di.uniroma1.it.`

‡Dipartimento di Matematica, Università degli Studi di Roma "Tor Vergata". `E-mail: clementi@mat.uniroma2.it.`

where $\texttt{dist}(v, w)$ is the Euclidean distance between v and w, while η is a constant that, without loss of generality, can be fixed to 1.

In several previous theoretical works [1, 10, 17, 22], it is assumed that nodes can arbitrarily vary their transmission range over the set $\{\texttt{dist}(v, w) \,|\, v, w \in V\}$. However, in some network models (like sensor networks), the adopted technology allows to have only few possible transmission range values. For this reason, we will assume that nodes have the ability to choose their transmission range from a finite set $\Gamma = \{0, r_1, r_2 \ldots, r_k\}$ (with $0 < r_1 < r_2 < \ldots < r_k$) that depends on the particular adopted technology (see [7, 8, 24]). Clearly, the maximal range value r_k in Γ must be sufficiently large to guarantee that at least one feasible solution exists. Further technical constraints on Γ will be given and discussed in Section 2.

1.1 Range assignments in ad-hoc wireless networks

A fundamental class of problems, underlying any phase of a dynamic resource allocation algorithm in ad-hoc wireless networks, is the one known as *range-assignment problems*. Given a specific graph-connectivity property Π, the objective of these problems is to find a transmission range assignment $r : V \to \Gamma$ such that: *(i)* r induces a communication graph satisfying Π, and *(ii)* its overall *cost*

$$\mathsf{cost}(r) = \sum_{v \in V} r(v)^2$$

required to deploy the assignment [17, 22], is minimized.
Several research works [1, 10, 17] have been devoted to the case where Π requires the communication graph to contain a directed spanning tree rooted at a given source $s \in V$ (a broadcast tree from s). The relevance of this problem, denoted as Min Energy Broadcast, is due to the fact that any communication graph satisfying the above property allows the source to perform a *broadcast* operation. Broadcast is a task initiated by the source aiming to send a message to all nodes. This task constitutes a fundamental operation in real life multi-hop wireless networks [2, 3, 17]. Min Energy Broadcast is known to be NP-hard even when $|\Gamma| = 3$ and r_1 is a small positive constant [10]. A series of constant-factor approximation algorithms are available in the literature (see, e.g., [1, 4, 10, 19]). The best known approximation factor is close to 4 and it is given in [6]. A more general version of Min Energy Broadcast is studied in [5], where a non-uniform *node efficiency* function $e : V \to \mathcal{R}^+$ is considered. Hence, the energy cost required to transmit from node v to w is given by $\texttt{dist}(v, w)^2/e(v)$. This non-symmetric version of Min Energy Broadcast seems to be harder: the best known algorithm is given in [5] and yields approximation ratio $\Theta(\log n)$.

1.2 The Max LifeTime problem

The Min Energy Broadcast problem does not consider some important ad-hoc wireless network scenarios where nodes are equipped with batteries of limited charge: the goal here is to maximize the number of broadcast operations. This important range-assignment problem has been first analytically studied in [5] and it is the subject of our paper.

Time is divided in *(time) periods*. Period t is devoted to broadcast the t-th message from the source s. All nodes are initially equipped with the same battery charge $B > 0$.
A *range-assignment schedule* $\mathcal{S}$ is a sequence of range assignment $\{r_t : V \to \Gamma, t = 1, \ldots, m\}$.

The *length* m of a range-assignment schedule is the number of periods. At every period t, the battery charge of each node v is reduced by amount $\beta\, r_t(v)^2$, where $r_t(v)$ denotes the range assigned to node v during t and $\beta > 0$ is a fixed constant depending on the adopted technology.

So, a range-assignment schedule is said to be *feasible* if, at any period t, r_t yields a broadcast tree from s and, for any $v \in V$, it holds that

$$\sum_{t=1}^{m} \beta r_t(v)^2 \leq B$$

In this paper, we assume $\beta = 1$, however, all our results can be easily extended to any $\beta > 0$. The MAX LIFETIME problem requires to find a feasible range-assignment schedule of maximal *length*.

In [5], MAX LIFETIME is shown to be NP-hard. In the same paper, by means of a rather involved reduction to MIN ENERGY BROADCAST with non uniform node efficiency, a polynomial-time algorithm is provided, yielding approximation ratio $\Theta(\log n)$. This positive result also holds when the initial node battery charges are not uniform.

A static version of MAX LIFETIME has been studied in [21]: the broadcast tree is fixed during the entire schedule and the quality of solutions returned by the MST-based algorithm is investigated. Such results and techniques are not useful for solving MAX LIFETIME problem, as the broadcast tree may change at each period.

Several other problems concerning network lifetime have been studied in the literature [7, 8, 21]. Their definitions vary depending on the particular node technology (i.e., fixed or adjustable node power) and on the required connectivity or covering property. However, both results and techniques (mostly of them being experimental) are not related to ours.

1.3 Our Results

To the best of our knowledge, previous *analytical* results on MAX LIFETIME concern worst-case instances only. Some *experimental* studies on MIN ENERGY BROADCAST have been done on *random geometric instances* [11, 19]. Such input distributions turn out to be very important in the study of range-assignment problems. On one hand, they represent the most natural random instance family where greedy heuristics (such as the MST-based one, see [17]) have a *bad* behaviour [19]. On the other hand, random geometric distributions provide a good model for *well-spread* networks located on 2-dimensional regions [7, 8, 17, 21].

We study MAX LIFETIME in random geometric instances of arbitrary size: the set V is formed by n nodes selected uniformly and independently at random from the 2-dimensional square of side length $\lfloor\sqrt{n}\rfloor$. Such instances will be simply denoted as *random sets*. Notice that the maximal Euclidean distance between two nodes in random sets is $\sqrt{2n}$, so the maximal range value r_k can be assumed to be at most $\sqrt{2n}$.

A natural and important open question is to establish whether efficiently-constructible range-assignment schedules exist for MAX LIFETIME having *provably-good* length on random sets. Moreover, the design of efficient *distributed* implementations of such schedules is of particular relevance in ad-hoc wireless networks.

To this aim, as a first step, we provide an upper bound on the length of an optimal range-assignment schedule $\mathcal{S}$ for *any* finite set V in the 2-dimensional plane. Notice that this upper bound holds for any instance, not only for random sets. When V is a random set we present an efficient centralized algorithm that, with high probability, returns a feasible schedule of length which is not smaller than 1/12 of the optimum. Here and in the sequel, the term *with high probability* means that the event holds with probability at least $1 - 1/n^c$ for some constant $c > 0$.

We then exploit our algorithm in order to design a fully distributed protocol for MAX LIFETIME. The protocol acts in discrete *time slots*[1] and assumes that every node initially knows n, its unique label in $[1, \ldots, n]$ (our results remain valid even if labels are chosen in $[1, \ldots, N]$, where $N \in O(n)$), and its Euclidean position.

This assumption is reasonable in *static* ad-hoc wireless networks since the node position can be either stored in the node during the deployment phase or it can be locally computed using a GPS system in a set-up phase. This operation is not too expensive in terms of energy consumption since it is performed only once during the set-up phase. We then show that the resulting schedule is equivalent to the one yielded by the centralized version and hence, when applied to random sets, it achieves, with high probability, constant approximation ratio as well. We thus get the first distributed algorithm for MAX LIFETIME having provably good performance.
We remark that in the analysis of our protocol we consider *both* the costs due to the *construction* of the range-assignment schedule and that due to its *use* for the broadcast operations. Furthermore, our protocol is designed to take care about message collisions yielded by the interference problems, thus no cost is hidden in the analysis.

The MAX LIFETIME problem does not consider the goal of minimizing the *completion time* of the broadcast operations, i.e., the number of time slots required to get all the nodes informed (a node is said to be *informed* if it has received the source message). However, the completion time is clearly an important measure to evaluate the efficiency of a solution. We thus provide an analysis of the *amortized* completion time of each broadcast operation yielded by our protocol. If the length of the range-assignment schedule generated by the protocol is T, then the amortized completion time is given by the overall number of elapsed time slots divided by T. It turns out that our protocol has amortized completion time

$$O\left(\frac{r_2 n\sqrt{n}}{T} + r_2^2 + \frac{\sqrt{n}}{r_2}\right).$$

Assume that $r_2 \in \Gamma$ is close to the *connectivity threshold* of *random geometric graphs* [16, 20, 25, 26], i.e., $r_2 = \Theta(\sqrt{\log n})$. Then, the worst scenario for our protocol is when the initial battery charge B is very small so that T is small as well. Indeed, if $T \in O(1)$, from the previous formula we get an amortized completion time $O(n\sqrt{n \log n})$, which is a factor $\sqrt{n \log n}$ larger than the best known distributed broadcasting time, i.e., $O(n)$ [16]. However, this protocol does not take into account *node energy* costs and, thus, the lifetime of the network. Our protocol, instead, trades completion time of each broadcast operation with global network lifetime. This fact clearly arises whenever B is large enough to allow $T \in \Omega(\sqrt{n})$ number of broadcast operations: in this case, we get $O(n\sqrt{\log n})$ amortized completion time, which is very close to the completion time of the best known distributed broadcast algorithm.

2 Preliminaries

A *random set* V is formed by n nodes selected uniformly and independently at random from the square Q of side length $\sqrt{n}$. The source node s can be any node in V. The length of a maximum feasible range-assignment schedule (in short, schedule) for an input (V, s) is denoted as $\texttt{opt}(V, s)$.

Given a set V of n nodes in the 2-dimensional Euclidean plane and a positive real r, the *disk graph* $G(V, r)$ is the symmetric graph where edge (u, v) exists iff $\texttt{dist}(v, w) \leq r$. When V

[1]Thus, periods are in turn divided in discrete time slots: a full description of the distributed model is given in Section 5.

is a random set, the resulting disk graph distribution is known as *geometric random graphs*, an important and deeply studied model in wireless networks [16, 20, 25, 26]. It is known that, for sufficiently large n, a random geometric graph $G(V, r)$ is connected with high probability if and only if $r \geq \mu\sqrt{\log n}$, where $\mu = 1+\epsilon$ for any constant $\epsilon > 0$ [20, 25, 26]. The value $\mathtt{CT}(n) = \mu\sqrt{\log n}$ is known as the *connectivity threshold* of random geometric graphs.

Notice that the connectivity threshold grows when the surface of the square grows as a consequence of the assumption of having a fixed node density for squares of any size. We could equivalently consider transmitting ranges of nodes as fixed, increasing node density while the area of the square grows in order to maintain the network connected with high probability. In this paper we adopt the former approach to simplify calculations, nevertheless our results can be easily translated to suit the latter approach and thus handle networks of growing sizes where nodes have fixed ranges.

Assumptions on range set Γ. We recall that $\Gamma = \{0, r_1, r_2 \ldots, r_k\}$ is such that $0 < r_1 < r_2 < \ldots < r_k \leq \sqrt{2n}$. In addition we assume that $1 \leq r_1 < \mathtt{CT}(n)$. This condition is motivated by our choice of studying random sets. Indeed, define C_s as the connected component containing s in the disk graph $G(V, r_1)$. If $r_1 \geq \mathtt{CT}(n)$ then MAX LIFETIME on random sets admits a trivial schedule which is, with high probability, optimal: Since the source must transmit in every period with range at least r_1 and C_s, with high probability, contains all nodes, then an optimal schedule is obtained by assigning range r_1 to all other nodes at every period. This motivates our assumption on r_1.
The other values in Γ can be arbitrarily chosen, provided that all of them are not smaller than $\mathtt{CT}(n)$ and at least one of them is larger than $2\sqrt{2}c\sqrt{\log n}$, where $c > \mu$ is a small constant that will be defined later in Lemma 4.1. Informally speaking, we require that at least one value in Γ is a bit larger than the connectivity threshold. This is reasonable and relevant in energy problems related to random geometric wireless networks since this value is the *minimal* one achieving global connectivity with high probability. Further discussion on such assumptions can be found in Section 6.

3 The upper bound

In this section, we provide an upper bound on the length of any feasible range-assignment schedule for a set V.

Lemma 3.1 *Given a set V and a source $s \in V$, it holds that $\mathtt{opt}(V, s) \leq B/r_1^2$. Furthermore, if the size k_1 of C_s is less than n, then*

$$\mathtt{opt}(V, s) \; \leq \; \min\left\{\frac{B}{r_1^2} \; , \; \frac{B}{r_2^4}\left(k_1 r_2^2 + r_1^2 - k_1 r_1^2\right)\right\}.$$

Proof. Since the source must transmit with range at least r_1 at any period, the first upper bound follows easily.

If $k_1 < n$ then consider any feasible range-assignment schedule $\mathcal{S}$. Let l_1 and l_2 be the number of periods where the source transmits with range r_1 and with range *at least* r_2, respectively. It must hold that

$$l_1 r_1^2 + l_2 r_2^2 \leq B.$$

Since $k_1 < n$ then, in each of the l_1 periods of $\mathcal{S}$, there is at least one node in $C_s - \{s\}$ having radius at least r_2. This yields

$$l_1 r_2^2 \leq (k_1 - 1)B.$$

The maximum value of $l_1 + l_2$ is achieved when $l_1 = (k_1 - 1)B/r_2^2$ and $l_2 = B/r_2^2 - (k_1 - 1)Br_1^2/r_2^4$. As $l_1 + l_2$ is the length of the schedule, we obtain the following upper bound on the number of periods of $\mathcal{S}$.

$$l_1 + l_2 \leq \min\left\{\frac{B}{r_1^2}, \frac{B}{r_2^4}\left(k_1 r_2^2 + r_1^2 - k_1 r_1^2\right)\right\}.$$

$\square$

Notice that if V is a random set then, since $r_1 < \mathtt{CT}(n)$, it holds with high probability $k_1 < n$.

4 The Algorithm

In this section we present a simple and efficient algorithm for Max LifeTime and then we analyze its performance. For the sake of simplicity, we restrict ourselves to the case $r_2 \geq 2\sqrt{2}c\sqrt{\log n}$. Nevertheless, it is easy to extend all our results to the more general assumption described in Section 2.

The following algorithm partitions Q into square cells and selects, for every period, a set of *pivots*, i.e., nodes having assigned range r_2. Each set of pivots is responsible for spreading the message of its period. This message is delivered to one of the pivots from the source by means of transmissions with range r_1, thus exploiting the subgraph C_s. Observe that we here assume every node knows the positions of all the other nodes and the cell partition: the algorithm is thus centralized.

Algorithm 1 : BS (Broadcast Schedule)

Input: Set $V \subseteq Q$ of n nodes; a source $s \in V$; a battery charge $B > 0$; the range set $\Gamma = \{0, r_1, r_2 \ldots, r_k\}$.
Output: A range-assignment schedule $\mathcal{S}$.

Partition Q into square *cells* of side length $r_2/(2\sqrt{2})$;
for any cell Q_j, let V_j be the set of nodes in Q_j;
construct an arbitrary ordering in V_j;
let C_s be the connected component in $G(V, r_1)$ that contains s;
if $|C_s| \leq r_2^2$ **then**
  $W_s \leftarrow C_s$;
else
  $W_s \leftarrow$ any connected subgraph of C_s s.t. $|W_s| = r_2^2$ and $s \in W_s$;
construct an arbitrary ordering of W_s;
for any period $t = 1, \ldots$ **do**
  if node with index t mod $|W_s|$ in W_s has remaining battery charge at least r_2^2 **then**
    it is selected as *pivot* and range r_2 is assigned to it;
  else
    the algorithm stops;
  for any cell Q_j **do**
    if node with index t mod $|V_j|$ in Q_j has remaining battery charge at least r_2^2 **then**
      it is selected as *pivot* and range r_2 is assigned to it;
    else
      the algorithm stops;
  all nodes in W_s not selected yet have radius r_1;
  all nodes in $V \setminus W_s$ not selected yet have range 0.

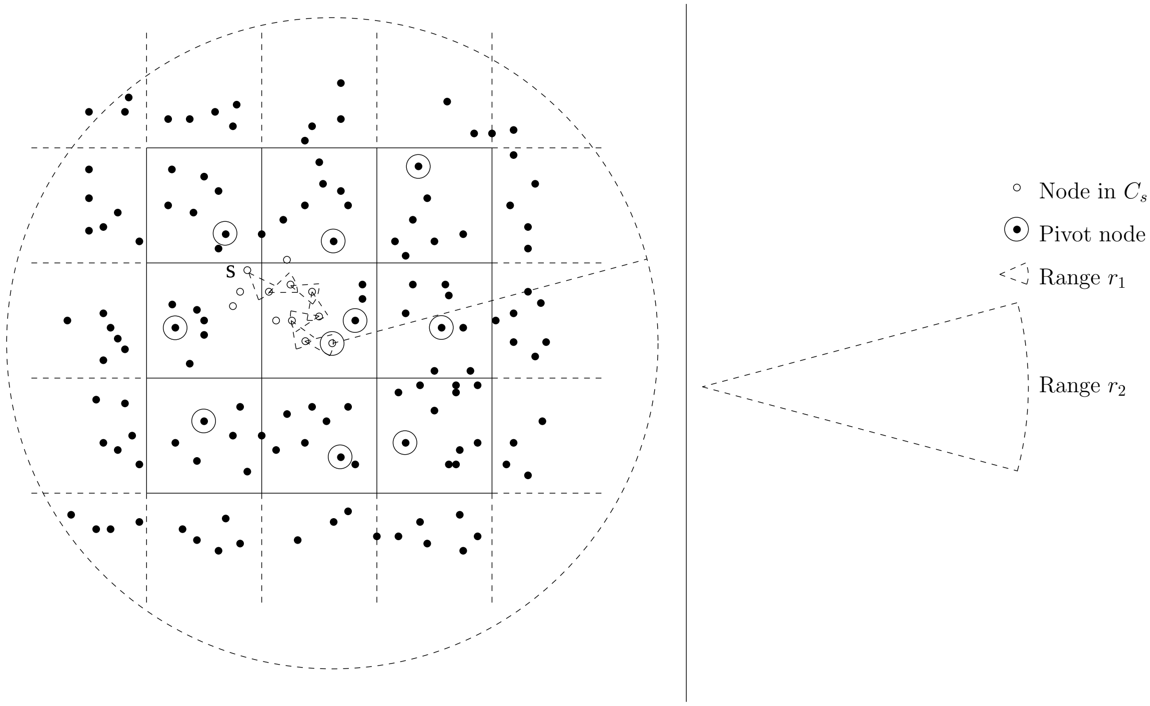

Figure 1: Execution example for Algorithm BS. The source message is sent through nodes in C_s to the first pivot, using range r_1. When a pivot transmits with range r_2, all nodes in its cell and in the neighboring cells receive the message.

See Figure 1 for an example of execution of Algorithm BS.

In order to analyze the performance of Algorithm BS, we will use the following lemma whose proof is a simple application of Chernoff's bound (an alternative proof can also be obtained from Lemma 1 of [18]).

Lemma 4.1 *There exist two positive constants, c and γ, such that the following holds. Given a random set $V \subseteq Q$ of n nodes, and a partition of Q into square* cells *of side length ℓ, where $c\sqrt{\log n} \leq \ell \leq \sqrt{n}$, every cell contains at least $\gamma\ell^2$ nodes with high probability. The constants can be set as $c = 12$ and $\gamma = 5/6$.*

Proof. Given a fixed cell, let X_i be the random variable that gets the value 1 if node i falls into the cell and 0 otherwise. Observe that: *(i)* the event $X_i = 1$ has probability ℓ^2/n, and *(ii)* the random variables X_i's, $1 \leq i \leq n$, are independent.
Let $X = \sum_{i=1}^{n} X_i$. The expected value of X, i.e. the average number of nodes in a cell of side ℓ, is $\mathrm{E}[X] = \ell^2$. By applying Chernoff's bound [23], we get

$$\Pr[X < \gamma\ell^2] < e^{-\ell^2(1-\gamma)^2/2}.$$

There are at most $\lfloor\sqrt{n}/\ell\rfloor^2 \leq n/\ell^2$ cells fully contained in Q, and thus the probability of having one such cell with less than $\gamma\ell^2$ nodes is at most $(n/\ell^2)e^{-\ell^2(1-\gamma)^2/2}$ by applying the Union Bound. As $\ell \geq c\sqrt{\log n}$, this probability is at most

$$e^{-c^2\log n(1-\gamma)^2/2+\ln\left(n/(c^2\log n)\right)} < e^{-c^2\log n(1-\gamma)^2/2+\log\left(n/(c^2\log n)\right)}.$$

By setting $\gamma = 5/6$ and $c \geq \sqrt{144} = 12$, we obtain

$$\Pr[X < \gamma \ell^2] < e^{-144/72 \log n + \log(n/(144 \log n))} \leq e^{-2 \log n + \log(n/(144 \log n))} < 1/n$$

$\square$

Theorem 4.2 *Let $V \subseteq Q$ be a random set of n nodes and $s \in V$ be any source node. Then, with high probability, the range-assignment schedule returned by Algorithm* BS *is feasible and it has length at least $\beta\, \texttt{opt}(V, s)$, where $\beta = 1/12$.*

Proof. Let us consider any period of the algorithm's schedule. The component W_s is not empty since it contains at least s. Hence, it contains a pivot which is connected to s by a path using ranges of size r_1 only. From Lemma 4.1, all cells are non empty with high probability. So, a pivot is selected in every cell with high probability. This implies that, with high probability, the set of pivots forms a strongly-connected subgraph whose r_2-disks cover all nodes in V. Moreover, Algorithm BS assigns, to every node, an energy power which is never larger than the current battery charge of the node. So the range-assignment schedule is feasible, with high probability.
We now evaluate the length T of the scheduling produced by Algorithm BS. Observe that T equals the index t of the *last* period performed by Algorithm BS on input (V, s).
Let w be any node in $V \setminus W_s$; then, from Lemma 4.1, in its cell there are at least $\gamma\, r_2^2/8$ nodes with high probability. So, w spends at most energy

$$\left(\frac{8T}{\gamma\, r_2^2}\right) r_2^2.$$

Hence, T can be any value such that

$$T \;\leq\; \frac{\gamma\, B}{8}. \tag{1}$$

During the schedule, every node v in W_s will have range either r_1 or r_2. Let $|W_s| = k$, then the energy spent by v is at most

$$\left(\frac{T}{k} + \frac{8T}{\gamma\, r_2^2}\right) r_2^2 + T r_1^2. \tag{2}$$

Indeed, in (2) we have considered that a node in W_s can have range r_2 because it has been selected as pivot either of its cell or of W_s.
Now, two cases may arise:

- If $k \geq \left(\frac{r_2}{r_1}\right)^2$, since $r_1 \geq 1$, from (2) the amount of spent energy is at most $2Tr_1^2 + 8/\gamma \leq Tr_1^2\,(2 + 8/\gamma)$. We require B to exceed the latter value, so, T can be any value such that

  $$T \;\leq\; \frac{B}{r_1^2\,(2 + 8/\gamma)}. \tag{3}$$

  Observe that every value T that satisfies (3) also satisfies (1). So T can assume value $\frac{B}{r_1^2(2+8/\gamma)}$ and, from Lemma 3.1, we have that

  $$T \geq \frac{\texttt{opt}(V, s)}{2 + 8/\gamma}.$$

- If $k < \left(\frac{r_2}{r_1}\right)^2$, according to the definition of W_s, we have $k = k_1$. From (2) and some simple calculations, the energy spent by $v \in W_s$ is at most

$$T \, \frac{r_2^4 + k_1 r_1^2 r_2^2 + (8/\gamma) k_1 r_2^2}{r_2^2 k_1 + r_1^2 - k_1 r_1^2}$$

where we used the fact that $r_1^2 - k_1 r_1^2 \leq 0$. Observe also that, since $k_1 < \left(\frac{r_2}{r_1}\right)^2$ and $r_1 \geq 1$, we have

$$k_1 r_1^2 r_2^2 + (8/\gamma) k_1 r_2^2 \leq r_2^4 \left(1 + \frac{8}{\gamma r_1^2}\right) \leq r_2^4 \left(1 + \frac{8}{\gamma}\right).$$

It thus follows that the energy spent by v is at most

$$T \frac{r_2^4 (2 + 8/\gamma)}{r_2^2 k_1 + r_1^2 - k_1 r_1^2}$$

Hence, T can be any value such that

$$T \leq \frac{r_2^2 k_1 + r_1^2 - k_1 r_1^2}{r_2^4 (2 + 8/\gamma)} B. \tag{4}$$

Similarly to the previous case, every value T that satisfies (4), it satisfies (1) as well. Finally, by combining (4) and Lemma 3.1, we get again

$$T \;\geq\; \frac{\texttt{opt}(V, s)}{2 + 8/\gamma}$$

So, the theorem is proved for $\beta = 1/(2 + 8/\gamma) > 1/12$. □

We conclude this section by observing that, when every node has full knowledge of the node positions, the time complexity of Algorithm BS is $O(r_2^2 + T \cdot (\sqrt{n}/r_2))$. Indeed, the organization of the nodes according to the cell partition and the construction of an arbitrary order of the nodes in every cell require linear time. The construction of W_s can be performed by a standard graph search, so it can be done in $O(n r_2^2)$ time. Finally, for each period t, constant time is required to activate the right pivot in every cell. The activation of those cells having the same r_2-hop distance from the source can be done in parallel. Since the maximum hop-distance from the source to any cell is $O(\sqrt{n}/r_2)$, we get the bound.

5 The distributed version

In this section, we present the distributed version of Algorithm BS.
According to the standard radio communication model [2, 13, 15], we assume that nodes act in discrete uniform *time slots* and are non spontaneous (but the source, the other nodes are activated when they get the source message). However, we assume a *weaker*, *local* synchronous model: if, at a given time slot t, the range of a message transmission covers a cell, then, at time slot $t+1$, the

nodes of that cell are activated and, so, they will agree on the same time. We assume that every node v knows the number n of points, the range Γ, its own unique label, and its relative coordinates in the square grid Q.

Our protocol is based on the same partition in cells of Algorithm BS. Cells are numbered with consecutive integers.
The aim of our protocol is to replicate the behaviour of Algorithm BS in a distributed fashion. In order to do so, nodes need to acquire some knowledge of the network within minimal cost in terms of energy and, as we shall see, time costs.

Let $h(W_s)$ be the eccentricity of the source s in W_s, i.e. the maximum distance between s and a node in W_s. The t-th message sent by s is denoted as m_t. We assume that m_t contains the value of period t. The protocol is described in Algorithm 2.

Our protocol has the following properties that are a key-ingredient in the performance analysis.

Fact 5.1 *Even though they initially do not known each other, all nodes in the same cell are activated (and disactivated) at the same time slot, so their local counters share the same value at every time slot. Furthermore, after the first $(\gamma/8)r_2^2$ broadcast operations, all nodes in the same cell know the set P of pivots of their cell and the relative order of its elements.*

Proof. A node in a cell is activated when it receives a message from the pivot of a neighboring cell and is set inactive when it receives the message sent by the pivot of its cell. Each point in a cell is within distance r_2 from all points in the same cell and in the neighboring cells; moreover, the pivots transmit with range r_2. So activations and de-activations of nodes in a cell happen in the same time slots. This proves the first part of the claim.
The set of pivots is learnt by all nodes in a cell as a consequence of the transmissions made by the pivots. Since this set has size bounded by $(\gamma/8)r_2^2$, the proof of the claim is completed. □

More practically, the above claim implies that if $l_0 < l_1 < l_2 < \ldots < l_t$ are the labels of the nodes in a cell, then, during the first $(\gamma/8)r_2^2$ broadcast operations (i.e. periods), the pivot of the cell at period t will be the node having label l_t.

In order to evaluate the length of the broadcast schedule yielded by Algorithm DBS, observe that the distributed version performs, in parallel, two tasks: (1) it constructs a broadcast communication subgraph starting from the source and (2) transmits the source message along this subgraph to all nodes. In our next analysis, all node costs due to both the above tasks are taken into account: whenever a node transmits any message with range r, its battery charge is decreased by r^2.

The following lemma states the equivalence between the performance of Algorithm BS and that of Algorithm DBS.

Lemma 5.2 *Given a random set $V \subseteq Q$ and any source $s \in V$, if the length of the broadcast schedule yielded by Algorithm* BS *is T, then the length of the broadcast schedule yielded by Algorithm* DBS *is at least $T - 2$.*

Proof. Notice that the only difference in terms of power consumption between Algorithms BS and DBS lies in the Preprocessing phase required by the latter one. In that phase, at most two messages with range r_1 are sent by a node to discover W_s. Hence, in the worst case, the distributed version performs two broadcasts less than the centralized algorithm. Notice that, thanks to Fact 5.1, the *if branch* of the Broadcast procedure spends *time* instead of *power* in order to discover the set of Pivots of each cell. □

By combining the above lemma and Theorem 4.2, we easily get the following

Algorithm 2 : DBS (Distributed Broadcast Schedule)

Preprocessing: construction of $W_s \subseteq C_s$ such that $h(W_s) \le r_2^2$

One-to-All

Starting from s, use round robin and range transmission r_1 to inform all nodes in C_s that are at most within r_2^2 hops from s: such nodes will form W_s.
The one-to-all operation induces a spanning tree `Tree` of W_s rooted at s.

All-to-One

By a simple bottom-up process on `Tree` and using round robin on each level, s collects all node labels and the structure of `Tree`.

Initialization:

Every node sets a local counter `counter` $= -1$. Furthermore, each node has a local array P of length $(\gamma/8)r_2^2$ where it will store the ordered list of the first $(\gamma/8)r_2^2$ labels belonging to its own cell. This array is initially empty.
Observe that at the end of the Preprocessing phase, s has full knowledge of W_s.

Broadcast operations:

for $t = 0, 1, \dots$ /* periods */ **do**
  Execute Procedure BROADCAST(m_t)

Procedure BROADCAST(m_t)

Nodes in W_s only:

- s selects the (t mod $\min\{|W_s|, r_2^2\}$)-th node in W_s as pivot (range r_2 will be assigned to it);
  s transmits, with range r_1, $\langle m_t, P\rangle$ where P is the path in `Tree` from s to the pivot.
- When a node in W_s receives $\langle m_t, P\rangle$, it checks whether its label is the first in P. If this is the case, it transmits, with range r_1, $\langle m_t, P'\rangle$ where P' is the residual path to the pivot.
- When the selected pivot p of W_s receives $\langle m_t, P = (p)\rangle$, it transmits, with range r_2, $\langle m_t, i\rangle$ where i is the index of its cell.

All nodes: (included the ones in W_s)

- If ($t \le (\gamma/8)r_2^2$) then
  - When a node v receives for the first time $\langle m_t, i\rangle$ in the period t from the pivot of a neighbor cell i, it becomes active.
  - An active node, at every time slot, increments `counter` by one and checks whether its label is equal to the value of its `counter`. If this is the case, it becomes the pivot of its cell and transmits, with range r_2, $\langle m_t, i\rangle$ where i is the index of its cell.
  - When an active node in cell i receives $\langle m_t, i\rangle$, it (so the pivot as well) records in $P[t]$ the current value of counter c, i.e. the label of the pivot, and becomes inactive.
- else (i.e. $(t > (\gamma/8)r_2^2)$)
  - When a node v receives for the first time $\langle m_t, i\rangle$ in period t from the pivot of a neighbor cell i, it checks if its label is equal to $P[t \bmod (\gamma/8)r_2^2]$. If this is the case, it becomes the pivot of its cell and transmits, with range r_2, $\langle m_t, j\rangle$ where j is the index of its cell.

Corollary 5.3 *Let $V \subseteq Q$ be a random set of n nodes and $s \in V$ be any source node. Then, with high probability, the range-assignment schedule yielded by Algorithm* DBS *is feasible and it has a length at least $\beta\, \mathtt{opt}(V,s) - 2$ where $\beta = 1/12$.*

We now evaluate message and time complexity of Protocol DBS.

Lemma 5.4 *The overall number of node transmissions (i.e. the message complexity) of Algorithm* DBS *is $O(|W_s| + T \cdot ((n/r_2^2) + r_2^2))$, where T is the length of the schedule.*

Proof. Observe that in the Preprocessing phase only nodes in C_s exchange messages. In particular, all nodes (the source as well) in C_s within r_2^2 hops from s send only one message; all other nodes, at hop-distance from 1 to $r_2^2 - 1$ from s, send two messages. It follows that the message complexity of the Preprocessing phase is $\Theta(|W_s|)$. During each broadcast, exactly one message per cell is sent (see Fact 5.1). As there are $O(n/r_2^2)$ cells, in each period an $O(n/r_2^2)$ number of messages are exchanged. The only further messages that are sent are the ones required to reach the elected pivot in W_s: the overall number of them is bounded by r_2^2. □

Theorem 5.5 *The overall number of time slots required by Algorithm* DBS *to perform T broadcast operations is*

$$O(r_2 n\sqrt{n} + T \cdot (r_2^2 + \sqrt{n}/r_2)).$$

Proof. For a single broadcast operation performed by Algorithm DBS, we define the *delay* of a cell as the number of time slots from its activation time to the selection of its pivot. Observe that the sum of delays introduced by a cell during the *first* $(\gamma/8)r_2^2$ broadcasts is at most $n - (\gamma/8)r_2^2$. Indeed, when a (new) pivot transmits, a new entry in the pivot-array is determined; while in every time slot there is silence, nodes in the cell learn that the node with the corresponding label is not in the cell. As each cell contains at least $(\gamma/8)r_2^2$ nodes with high probability, the delay introduced by each cell before completing the set is with high probability at most $n - (\gamma/8)r_2^2$. Once the set of pivots is completed (i.e., after the first $(\gamma/8)r_2^2$ periods), the delay of every cell becomes 0 for all the remaining broadcasts. Moreover, a broadcast can cross at most $O(\sqrt{n}/r_2)$ cells (as the side length of each cell is $\Theta(r_2)$, while the diameter of Q is $\Theta(\sqrt{n})$). By assuming the worst-scenario, i.e., a maximal length cell path (this length being $\Theta(\sqrt{n}/r_2)$) together with maximal cell delay can be found in each of the first $\min\{(\gamma/8)r_2^2, T\}$ broadcasts, we can bound the maximal overall delay with $O(r_2 n\sqrt{n})$ time slots.
In the Preprocessing phase, Algorithm DBS uses round robin to avoid collisions. During the All-to-One phase, each node needs to collect all messages from its children before sending a message to its parent in `Tree`. Hence, the whole phase is completed in $O(nr_2^2)$ time slots, as the height of `Tree` is bounded by r_2^2.
Finally, the number of time slots required by every broadcast without delays and Preprocessing time is $O(r_2^2 + \sqrt{n}/r_2)$, since r_2^2 is the upper bound on the height of `Tree` and the length of any path on the broadcast tree outside W_s is $O(\sqrt{n}/r_2)$.
By combining the three contributions, we get the theorem bound without considering collisions among pivots of adjacent cells. In order to avoid such collisions, we further organize Algorithm DBS into iterative phases: in every phase, only cells with not colliding pivot transmissions are active. Since the number of cells that can interfere with a given cell is constant, this further scheduling will increase the overall time of DBS by a constant factor only. This iterative process can be efficiently performed in a distributed way since every node knows n and its position, so it

knows its cell. □

From Theorem 5.5, the amortized completion time of a single broadcast operation performed by Algorithm DBS is

$$O\left(\frac{r_2 n\sqrt{n}}{T} + r_2^2 + \frac{\sqrt{n}}{r_2}\right).$$

Since our protocol returns an almost maximal number T of broadcast operations with high probability, unless the available battery charge of nodes is small, the analysis we made at the end of Subsection 1.3 on the amortized completion time of Algorithm DBS is likely to fall in a scenario in which T is large enough to significantly shrink the gap between our distributed algorithms and the best known distributed broadcasting time [16].

6 Open Problems

In this paper, we provided efficient solutions for the Max LifeTime problem on random sets. Further interesting future studies should address other basic operations such as, for instance, the gossiping operation which is known to be NP-hard as well [5]. A more technical problem, left open by our work, is the study of Max LifeTime when Γ contains more than one *positive* values *smaller* than the connectivity threshold $\mathtt{CT}(n)$ of random geometric graphs. This case seems to be very hard since it concerns the size and the structure of the connected components of such random graphs *under* the connectivity threshold [20, 25].
Finally, we emphasize that, after the presentation of the conference version of this work, a new protocol for Min Energy Broadcast has been given in [9]. This new protocol is inspired by ours and achieves provably-good performances on random-grid instances yielded by non-uniform node distributions.

References

[1] C. Ambuehl. An optimal bound for the MST algorithm to compute energy efficient broadcast trees in wireless networks. In Proc. of *ICALP'05*, 1139–1150, 2005.

[2] R. Bar-Yehuda, O. Goldreich, and A. Itai. On the time-complexity of broadcast in multi-hop radio networks: An exponential gap between determinism and randomization. *JCSS*, 45, 104–126, 1992.

[3] R. Bar-Yehuda, A. Israeli, and A. Itai. Multiple communication in multi-hop radio networks, *SICOMP*, 22 (4), 875–887, 1993.

[4] G. Calinescu, X.Y. Li, O. Frieder, and P.J. Wan. Minimum-Energy Broadcast Routing in Static Ad Hoc Wireless Networks. In Proc. of *IEEE INFOCOM'01*, 1162–1171, April 2001.

[5] G. Calinescu, S. Kapoor, A. Olshevsky, A. Zelikovsky. Network lifetime and power assignment in ad hoc wireless networks. In Proc. of *ESA03*, LNCS, 2832, 114–126, 2003.

[6] I. Caragiannis, M. Flammini, and L. Moscardelli. An exponential improvement on the MST heuristic for the minimum energy broadcast problem. In Proc. of *ICALP'07*, LNCS, 4596, 447–458, 2007.

[7] M. Cardei and D-Z Du. Improving wireless sensor network lifetime through power organization. *Wireless Networks*, 11, 333–340, 2005.

[8] M. Cardei, J. Wu, and M. Lu. Improving network lifetime using sensors with adjustable sensing ranges. *Int. J. Sensor Networks*, 1 (1/2), 41–49, 2006.

[9] T. Calamoneri, A. Clementi, A. Monti, G. Rossi, and R. Silvestri. Minimum-energy broadcast in random-grid ad-hoc networks: approximation and distributed algorithms. In Proc. of *11-th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '08)*, ACM Press, 2008.

[10] A. Clementi, P. Crescenzi, P. Penna, G. Rossi and P. Vocca. On the Complexity of Computing Minimum Energy Consumption Broadcast Subgraphs. In Proc. of 18th *STACS*, LNCS 2010, 121–131, February 2001 (full version available at `www.dia.unisa.it/~penna`).

[11] A. Clementi, G. Huiban, P. Penna, G. Rossi, Y. C. Verhoeven. On the Approximation Ratio of the MST-based Heuristic for the Energy-Efficient Broadcast Problem in Static Ad-Hoc Radio Networks. In Proc. of *IPDPS'03*, 222, 2003.

[12] A. E. F. Clementi, A. Monti, and R. Silvestri. Distributed broadcast in radio networks of unknown topology, *Theor. Comput. Sci.*, 302(1-3), 337–364, 2003.

[13] M. Chrobak, L. Gasieniec, and W. Rytter. Fast broadcasting and gossiping in radio networks. *J. Algorithms*, 43(2), 177–189, 2002.

[14] W. Chu, C.J. Colbourn, and V.R. Syrotiuk. The effects of synchronization on topology transparent scheduling. *Wireless Networks*, 12, 681–690, 2006.

[15] A. Czumaj, W. Rytter. Broadcasting algorithms in radio networks with unknown topology. *J. Algorithms*, 60(2), 115–143, 2006.

[16] A. Dessmark and A. Pelc. Broadcasting in geometric radio networks. *J. Discrete Algorithms*, 5(1), 187–201, 2007.

[17] A. Ephremides, G.D. Nguyen, and J.E. Wieselthier. On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks. In Proc. of *IEEE INFOCOM'00*, 585–594, 2000.

[18] A. D. Flaxman, A. M. Frieze, and J. C. Vera. On the average case performance of some greedy approximation algorithms for the uncapacitated facility location problem. In Proc. of *ACM STOC'05*, 441–449, 2005.

[19] M. Flammini, A. Navarra, and S. Perennes. The Real Approximation Factor of the MST Heuristic for the Minimum Energy Broadcast. In Proc. of *WEA'05*, 22–31, 2005.

[20] P. Gupta and P.R. Kumar. Critical power for asymptotic connectivity in wireless networks. In *Stochastic Analysis, Control, Optimization and Applications*. Birkhauser, Boston, MA, 547–566, 1999.

[21] I. Kang and R. Poovendran. Maximizing network lifetime of wireless broadcast ad hoc networks. *J. of ACM Mobile Networks and Applications*, 10(6), 879–896, 2005.

[22] L. M. Kirousis, E. Kranakis, and D. Krizanc, and A. Pelc. Power Consumption in Packet Radio Networks. *Theoretical Computer Science*, 243, 289–305, 2000.

[23] M. Mitzenmacher and E. Upfal. *Probability and Computing*, Cambridge University Press, 2005.

[24] K. Pahlavan and A. Levesque. *Wireless Information Networks*. Wiley-Interscience, 1995.

[25] M. Penrose. *Random Geometric Graphs*. Oxford University Press, 2003.

[26] P. Santi and D. M. Blough. The Critical Transmitting Range for Connectivity in Sparse Wireless Ad Hoc Networks. *IEEE Trans. on Mobile Computing*, 2, 25–39, 2003.