Radio Network Clustering from Scratch

Fabian Kuhn, Thomas Moscibroda, Roger Wattenhofer
{kuhn,moscitho,wattenhofer}@inf.ethz.ch

Department of Computer Science, ETH Zurich, 8092 Zurich, Switzerland

Abstract. We propose a novel randomized algorithm for computing a dominating set based clustering in wireless ad-hoc and sensor networks. The algorithm works under a model which captures the characteristics of the set-up phase of such multi-hop radio networks: asynchronous wake-up, the hidden terminal problem, and scarce knowledge about the topology of the network graph. When modelling the network as a unit disk graph, the algorithm computes a dominating set in polylogarithmic time and achieves a constant approximation ratio.

1 Introduction

Ad-hoc and sensor networks are formed by autonomous nodes communicating via radio, without any additional infrastructure. In other words, the communication infrastructure is provided by the nodes themselves. When being deployed, the nodes initially form an *unstructured* radio network, which means that no reliable and efficient communication pattern has been established yet. Before any reasonable communication can be carried out, the nodes must establish a media access control (MAC) scheme which provides reliable point-to-point connections to higher-layer protocols and applications. The problem of setting up an initial structure in radio networks is of great importance in practice. Even in a single-hop ad-hoc network such as Bluetooth and for a small number of devices, the initialization tends to be slow. Clearly, in a multi-hop scenario with many nodes, the time consumption for establishing a communication pattern increases even further. In this paper, we address this *initialization process*.

One prominent approach to solving the problem of bringing structure into a multi-hop radio network is a *clustering*, in which each node in the network is either a cluster-head or has a cluster-head within its communication range (such that cluster-heads can act as coordination points for the MAC scheme) [1, 4, 6]. When we model a multi-hop radio network as a graph $G = (V, E)$, this clustering can be formulated as a classic graph theory problem: In a graph, a *dominating set* is a subset $S \subseteq V$ of nodes such that for every node v, either a) $v \in S$ or b) $v' \in S$ for a direct neighbor v' of v. As it is desirable to compute a dominating set with few *dominators*, we study the *minimum dominating set* (MDS) problem which asks for a dominating set of minimum cardinality.

The computation of dominating sets for the purpose of structuring networks has been studied extensively and a variety of algorithms have been proposed, e.g. [4, 7, 9, 10, 12]. To the best of our knowledge, all these algorithms operate on an existing MAC layer, providing point-to-point connections between neighboring nodes. While this is valid in structured networks, it is certainly an improper assumption for the initialization phase. In fact, by assuming point-to-point connections, many vital problems arising in unstructured networks (collision detection, asynchronous wake-up, or the hidden terminal problem) are simply abstracted away. Consequently, none of the existing dominating set algorithms helps in the initialization process of such networks.

We are interested in a simple and practical algorithm which quickly computes a clustering from *scratch*. Based on this initial clustering, the MAC layer can subsequently be established. An unstructured multi-hop radio network can be modelled as follows:

- The network is *multi-hop*, that is, there exist nodes that are not within their mutual transmission range. Being multi-hop complicates things since some of the neighbors of a sending node may receive a transmission, while others are experiencing interference from other senders and do not receive the transmission.
- The nodes do not feature a reliable *collision detection* mechanism [2, 5, 8]. In many scenarios not assuming any collision detection mechanism is realistic. Nodes may be tiny sensors in a sensor network where equipment is restricted to the minimum due to limitations in energy consumption, weight, or cost. The absence of collision detection includes sending nodes, i.e., the sender does not know whether its transmission was received successfully or whether it caused a collision.
- Our model allows nodes to wake-up *asynchronously*. In a multi-hop environment, it is realistic to assume that some nodes wake up (e.g. become deployed, or switched on) later than others. Consequently, nodes do not have access to a global clock. Asynchronous wake-up rules out an ALOHA-like MAC schemes as this would result in a linear runtime in case only one single node wakes up for a long time.
- Nodes have only limited knowledge about the total number of nodes in the network and no knowledge about the nodes' distribution or wake-up pattern.

In this paper, we present a randomized algorithm which computes an asymptotically optimal clustering for this harsh model in polylogarithmic time only. Section 2 gives an overview over relevant previous work. Section 3 introduces our model as well as some well-known facts. The algorithm is developed and analyzed in Sections 4 and 5.

2 Related Work

The problem of finding a minimum dominating set was proven to be NP-hard. Furthermore, it has been shown in [3] that the best possible approximation ratio for this problem is $\ln \Delta$ where Δ is the highest degree in the graph, unless NP has deterministic $n^{O(\log \log n)}$-time algorithms. For unit disk graphs, the problem remains NP-hard, but constant factor approximations are possible. Several distributed algorithms have been proposed, both for general graphs [7, 9, 10] and the Unit Disk Graph [4, 12]. All the above algorithms assume point-to-point connections between neighboring nodes and are thus unsuitable in the context of initializing radio networks.

A model similar to the one used in this paper has previously been studied in the context of analyzing the complexity of broadcasting in multi-hop radio networks, e.g. [2]. A striking difference to our model is that throughout the literature on broadcast in radio networks, *synchronous wake-up* is considered, i.e. all nodes have access to a global clock and start the algorithm simultaneously. A model featuring asynchronous wake-up has been studied in recent papers on the *wake-up problem* in single-hop networks [5, 8]. In comparison to our model, these papers define a much *weaker notion of asynchrony*. Particularly, it is assumed that sleeping nodes are *woken up* by a successfully transmitted message. In a single-hop network, the problem of waking up all nodes thus reduces to analyzing the number of time-slots until one message is successfully transmitted. While this definition of asynchrony leads to theoretically interesting problems and algorithms, it does not closely reflect reality.

3 Model

We model the *multi-hop* radio network with the well known *Unit Disk Graph* (UDG). In a UDG $G = (V, E)$, there is an edge $\{u, v\} \in E$ iff the Euclidean distance between u and v is at most 1. Nodes may wake up *asynchronously* at any time. We call a node *sleeping* before its wake-up, and *active* thereafter. Sleeping nodes can neither send nor receive any messages, regardless of their being within the transmission range of a sending node. Nodes do not have any a-priori knowledge about the topology of the network. They only have an upper bound $\hat{n}$ on the number of nodes $n = |V|$ in the graph. While n is unknown, all nodes have the same estimate $\hat{n}$. As shown in [8], without any estimate of n and in absence of a global clock, every algorithm requires at least time $\Omega(n/\log n)$ until one single message can be transmitted without collision.

While our algorithm does not rely on synchronized time-slots in any way, we do assume time to be divided into time-slots in the analysis section. This simplification is justified due to the trick used in the analysis of slotted vs. unslotted ALOHA [11], i.e., a single packet can cause interference in no more than two consecutive time-slots. Thus, an analysis in an "ideal" setting with synchronized time-slots yields a result which is only by a constant factor better as compared to the more realistic unslotted setting.

We assume that nodes have three independent communication channels Γ_1, Γ_2, and Γ_3 which may be realized with an FDMA scheme. In each time-slot, a node can either send or not send. Nodes do not have a *collision detection mechanism*, that is, nodes are unable to distinguish between the situation in which two or more neighbors are sending and the situation in which no neighbor is sending. A node receives a message on channel Γ in a time-slot only if *exactly one neighbor* has sent a message in this time-slot on Γ. A sending node does not know how many (if any at all!) neighbors have correctly received its transmission. The variables p_k and q_k denote the probabilities that node k sends a message in a given time-slot on Γ_1 and Γ_2, respectively. Unless otherwise stated, we use the term *sum of sending probabilities* to refer to the sum of sending probabilities on Γ_1. We conclude this section with two facts. The first was proven in [8] and the second can be found in standard mathematical textbooks.

Fact 1 *Given a set of probabilities* $p_1 \dots p_n$ *with* $\forall i : p_i \in [0, \frac{1}{2}]$, *the following inequalities hold:* $(1/4)^{\sum_{k=1}^{n} p_k} \ \leq \ \prod_{k=1}^{n} (1 - p_k) \ \leq \ (1/e)^{\sum_{k=1}^{n} p_k}$.

Fact 2 *For all* n, t, *with* $n \geq 1$ *and* $|t| \leq n$, $\ e^t \left(1 - t^2/n\right) \ \leq \ (1 + t/n)^n \ \leq \ e^t$.

Algorithm 1 Dominator Algorithm

decided := dominator := false;
upon wake-up do:
1: **for** $j := 1$ to $\delta \cdot \lceil \log \hat{n} \rceil$ by 1 **do**
2:   **if** message received in current time-slot **then** decided := true; **fi**
3: **end for**
4: **for** $j := \lceil \log \hat{n} \rceil$ to 0 by -1 **do**
5:   $p := 1/\left(2^{j+\beta}\right)$;
6:   **for** $i := 1$ to δ by 1 **do**
7:     $b_i^{(1)} := 0$; $b_i^{(2)} := 0$; $b_i^{(3)} := 0$;
8:     **if not** $\mathrm{decided}$ **then**
9:       $b_i^{(1)} := 1$, with probability p;
10:       **if** $b_i^{(1)} = 1$ **then** dominator := true;
11:       **else if** message received in current time-slot **then** decided := true;
12:       **fi**
13:     **end if**
14:     **if** $\mathrm{dominator}$ **then**
15:       $b_i^{(2)} := 1$, with probability q; $b_i^{(3)} := 1$, with probability $q/\log \hat{n}$;
16:     **end if**
17:     **if** $b_i^{(1)} = 1$ **then** send message on Γ_1 **fi**
18:     **if** $b_i^{(2)} = 1$ **then** send message on Γ_2 **fi**
19:     **if** $b_i^{(3)} = 1$ **then** send message on Γ_3 **fi**
20:   **end for**
21: **end for**
22: **if not** $\mathrm{decided}$ **then** dominator := decided := true; **fi**
23: **if** $\mathrm{dominator}$ **then**
24:   **loop**
25:     send message on Γ_2 and Γ_3, with probability q and $q/\log \hat{n}$, respectively;
26:   **end loop**
27: **end if**

4 Algorithm

A node starts executing the dominator algorithm (Algorithm 1) upon waking up. In the first phase (lines 1 to 3), nodes wait for messages (on all channels) without sending themselves. The reason is that nodes waking up late should not interfere with already existing dominators. Thus, a node first listens for existing dominators in its neighborhood before actively trying to become dominator itself.

The main part of the algorithm (starting in line 4) works in rounds, each of which contains δ time-slots. In every time-slot, a node sends with probability p on channel Γ_1. Starting from a very small value, this sending probability p is doubled (lines 4 and 5) in every round. When sending its first message, a node becomes a dominator and, in addition to its sending on channel Γ_1, it starts sending on channels Γ_2 and Γ_3 with probability q and $q/\log n$, respectively. Once a node becomes a dominator, it will remain so for the rest of the algorithm's execution. For the algorithm to work properly, we must prevent the sum of sending probabilities on channel Γ_1 from reaching too high values. Otherwise, too many collisions will occur, leading to a large number of dominators. Hence, upon receiving its first message (without collision) on any channel, a node becomes *decided* and stops sending on Γ_1. Being decided means that the node is covered by a dominator and consequently, the node stops sending on Γ_1.

Thus, the basic intuition is that nodes, after some initial listening period, compete to become dominator by exponentially increasing their sending probability on Γ_1. Channels Γ_2 and Γ_3 then ensure that the number of further dominators emerging in the neighborhood of an already existing dominator is bounded.

The parameters q, β, and δ of the algorithm are chosen as to optimize the results and guarantee that all claims hold with high probability. In particular, we define $q := (2^{\beta} \cdot \lceil \log \hat{n} \rceil)^{-1}$, $\delta := \lceil \log(\hat{n}) / \log(503/502) \rceil$, and $\beta := 6$. The parameter δ is chosen large enough to ensure that with high probability, there is a round in which at least one competing node will send without collision. The parameter q is chosen such that during the "waiting time-slots", a new node will receive a message from an existing dominator. Finally, β maximizes the probability of a successful execution of the algorithm. The algorithm's correctness and time-complexity (defined as the number of time-slots of a node between wake-up and decision) follow immediately:

Theorem 1. *The algorithm computes a correct dominating set. Moreover, every node decides whether or not to become dominator in time* $\mathrm{O}(\log^2 \hat{n})$.

Proof. The first for-loop is executed $\delta \cdot \lceil \log \hat{n} \rceil$ times. The two nested loops of the algorithm are executed $\lceil \log \hat{n} \rceil + 1$ and δ times, respectively. After these two loops, all remaining undecided nodes decide to become dominator.

5 Analysis

In this section, we show that the expected number of dominators in the network is within $\mathrm{O}(1)$ of an optimal solution. As argued in Section 3, we can simplify the analysis by assuming all nodes operate in synchronized time-slots.

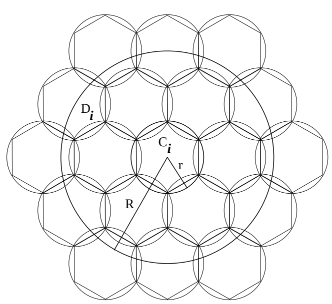

Fig. 1. Circles C_i and D_i

We cover the plane with circles C_i of radius $r = 1/2$ by a hexagonal lattice shown in Figure 1. Let D_i be the circle centered at the center of C_i having radius $R = 3/2$. It can be seen in Figure 1 that D_i is (fully or partially) covering 19 smaller circles C_j. Note that every node in a circle C_i can hear all other nodes in C_i. Nodes outside D_i are not able to cause a collision in C_i.

The proof works as follows. We first bound the sum of sending probabilities in a circle C_i. This leads to an upper bound on the number of collisions in a circle before at least one dominator emerges. Next, we give a probabilistic bound on the number of sending nodes per collision. In the last step, we show that nodes waking up being already covered do not become dominator.

All these claims hold with high probability. Note that for the analysis, it is sufficient to assume $\hat{n} = n$, because solving minimum dominating set for $n' < n$ cannot be more difficult than for n. If it were, the imaginary adversary controlling the wake-ups of all nodes could simply decide to let $n - n'$ sleep infinitely long, which is indistinguishable from having n' nodes.

Definition 1. *Consider a circle C_i. Let t be a time-slot in which a message is sent by a node $v \in C_i$ on Γ_1 and received (without collision) by all other nodes in C_i. We say that circle C_i* clears *itself in time-slot t. Let t_0 be the first such time-slot. We say that C_i* terminates *itself in time-slot t_0. For all time-slots $t' \geq t_0$, we call C_i* terminated.

Definition 2. *Let $s(t) := \sum_{k \in C_i} p_k$ be the sum of sending probabilities on Γ_1 in C_i at time t. We define the time slot t_i^j so that for the j^{th} time in C_i, we have $s(t_i^j - 1) < \frac{1}{2^\beta}$ and $s(t_i^j) \geq \frac{1}{2^\beta}$. We further define the Interval $\mathcal{I}_i^j := [t_i^j \ldots t_i^j + \delta - 1]$.*

In other words, t_i^0 is the time-slot in which the sum of sending probabilities in C_i exceeds $\frac{1}{2^\beta}$ for the first time and t_i^j is the time-slot in which this threshold is surpassed for the j^{th} time in C_i.

Lemma 1. *For all time-slots $t' \in \mathcal{I}_i^j$, the sum of sending probabilities in C_i is bounded by $\sum_{k \in C_i} p_k \ \leq \ 3/2^\beta$.*

Proof. According to the definition of t_i^j, the sum of sending probabilities $\sum_{k \in C_i} p_k$ at time $t_i^j - 1$ is less than $\frac{1}{2^\beta}$. All nodes which are active at time t_i^j will double their sending probability p_k exactly once in the following δ time-slots. Previously inactive nodes may wake up during that interval. There are at most n of such newly active nodes and each of them will send with the initial sending probability $\frac{1}{2^\beta \hat{n}}$ in the given interval. In $\mathcal{I}_i^j$, we get

$$\sum_{k \in C_i} p_k \ \leq \ 2 \cdot \frac{1}{2^\beta} + \sum_{k \in C_i} \frac{1}{2^\beta \hat{n}} \ \leq \ 2 \cdot \frac{1}{2^\beta} + \frac{n}{2^\beta \hat{n}} \ \leq \ \frac{3}{2^\beta}.$$

Using the above lemma, we can formulate a probabilistic bound on the sum of sending probabilities in a circle C_i. Intuitively, we show that before the bound can be surpassed, C_i does either clear itself or some nodes in C_i become decided such that the sum of sending probabilities decreases.

Lemma 2. *The sum of sending probabilities of nodes in a circle C_i is bounded by $\sum_{k \in C_i} p_k \ \leq \ 3/2^\beta$ with probability at least $1 - \mathrm{o}\left(\frac{1}{n^2}\right)$. The bound holds for all C_i in G with probability at least $1 - \mathrm{o}\left(\frac{1}{n}\right)$.*

Proof. The proof is by induction over all intervals $\mathcal{I}_i^j$ and the corresponding time-slots t_i^j in ascending order. Lemma 1 states that the sum of sending probabilities in C_i is bounded by $\frac{3}{2^\beta}$ in each interval $\mathcal{I}_i^j$. In the sequel, we show that in $\mathcal{I}_i^j$, the circle C_i either clears itself or the sum of sending probabilities falls back below $\frac{1}{2^\beta}$ with high probability. Note that for all time-slots t *not covered* by any interval $\mathcal{I}_i^j$, the sum of sending probabilities is below $\frac{1}{2^\beta}$. Hence, the induction over all intervals is sufficient to prove the claim.

Let $t' := t_i^0$ be the very first critical time-slot in the network and let $\mathcal{I}'$ the corresponding interval. If some of the active nodes in C_i receive a message from a neighboring node, the sum of sending probabilities may fall back below $\frac{1}{2^\beta}$. In this case, the sum does obviously not exceed $\frac{3}{2^\beta}$. If the sum of sending probabilities does not fall back below $\frac{1}{2^\beta}$, the following two inequalities hold for the duration of the interval $\mathcal{I}'$:

$$\frac{1}{2^\beta} \leq \sum_{k \in C_i} p_k \leq \frac{3}{2^\beta} : in\ C_i \tag{1}$$

$$0 \leq \sum_{k \in C_j} p_k \leq \frac{3}{2^\beta} : in\ C_j \in D_i, i \neq j. \tag{2}$$

The second inequality holds because t' is the very first time-slot in which the sum of sending probabilities exceeds $\frac{1}{2^\beta}$. Hence, in each $C_j \in D_i$, the sum of sending probabilities is at most $\frac{3}{2^\beta}$ in $\mathcal{I}'$. (Otherwise, one of these circles would have reached $\frac{1}{2^\beta}$ before circle C_i and t' is not the first time-slot considered).

We will now compute the probability that C_i clears itself within $\mathcal{I}'$. Circle C_i clears itself when exactly one node in C_i sends and no other node in $D_i \setminus C_i$ sends. The probability P_0 that no node in any neighboring circle $C_j \in D_i, j \neq i$ sends is

$$\begin{aligned} P_0 &= \prod_{\substack{C_j \in D_i \\ j \neq i}} \prod_{k \in C_j} (1 - p_k) \underset{\text{Fact 1}}{\geq} \prod_{\substack{C_j \in D_i \\ j \neq i}} \left(\frac{1}{4}\right)^{\sum_{k \in C_j} p_k} \\ &\underset{\text{Lemma 1}}{\geq} \prod_{\substack{C_j \in D_i \\ j \neq i}} \left(\frac{1}{4}\right)^{\frac{3}{2^\beta}} \geq \left[\left(\frac{1}{4}\right)^{\frac{3}{2^\beta}}\right]^{18}. \end{aligned} \tag{3}$$

Let P_{suc} be the probability that exactly one node in C_i sends:

$$\begin{aligned} P_{suc} &= \sum_{k \in C_i} \left(p_k \cdot \prod_{\substack{l \in C_i \\ l \neq k}} (1 - p_l) \right) \geq \sum_{k \in C_i} p_k \cdot \prod_{l \in C_i} (1 - p_l) \\ &\underset{\text{Fact 1}}{\geq} \sum_{k \in C_i} p_k \cdot \left(\frac{1}{4}\right)^{\sum_{k \in C_i} p_k} \geq \frac{1}{2^\beta} \cdot \left(\frac{1}{4}\right)^{\frac{1}{2^\beta}}. \end{aligned}$$

The last inequality holds because the previous function is increasing in $[\frac{1}{2^\beta}, \frac{3}{2^\beta}]$.

The probability P_c that exactly one node in C_i and no other node in D_i sends is therefore given by

$$P_c = P_0 \cdot P_{suc} \geq \left[\left(\frac{1}{4}\right)^{\frac{3}{2^\beta}}\right]^{18} \cdot \frac{1}{2^\beta} \left(\frac{1}{4}\right)^{\frac{1}{2^\beta}} \underset{\beta=6}{=} \frac{2^{9/32}}{256}.$$

P_c is a lower bound for the probability that C_i clears itself in a time-slot $t \in \mathcal{I}'$. The reason for choosing $\beta = 6$ is that this value maximizes P_c. The probability $\overline{P_{term}}$ that circle C_i does not clear itself during the entire interval is $\overline{P_{term}} \leq (1 - 2^{9/32}/256)^{\delta} \leq n^{-2.3} \in \mathrm{o}\big(n^{-2}\big)$. We have thus shown that within $\mathcal{I}'$, the sum of sending probabilities in C_i either falls back below $\frac{1}{2^\beta}$ or C_i clears itself.

For the induction step, we consider an arbitrary t_i^j. Due to the induction hypothesis, we can assume that all previous such time-slots have already been dealt with. In other words, all previously considered time-slots $t_{i'}^{j'}$ have either lead to a clearance of circle $C_{i'}$ or the sum of probabilities in $C_{i'}$ has decreased below the threshold $\frac{1}{2^\beta}$. Immediately after a clearance, the sum of sending probabilities in a circle C_i is at most $\frac{1}{2^\beta}$, which is the sending probability in the last round of the algorithm. This is true because only one node in the circle remains undecided. All others will stop sending on channel Γ_1. By Lemma 1, the sum of sending probabilities in all neighboring circles (both the cleared and the not cleared ones) is bounded by $\frac{3}{2^\beta}$ in $\mathcal{I}_i^j$ (otherwise, this circle would have been considered before t_i^j). Therefore, we know that the bounds (1) and (2) hold with high probability and the computation for the induction step is the same as for the base case t'.

Because there are n nodes to be decided and at most n circles C_i, the number of induction steps t_i^j is at most n. Hence, the probability that the claim holds for all steps is at least $\big(1 - \mathrm{o}\big(\frac{1}{n^2}\big)\big)^n \geq 1 - \mathrm{o}\big(\frac{1}{n}\big)$.

Using Lemma 2, we can now compute the expected number of dominators in each circle C_i. In the analysis, we will separately compute the number of dominators *before* and *after* the termination (i.e., the first clearance) of C_i.

Lemma 3. *Let C be the number of collisions (more than one node is sending in one time-slot on Γ_1) in a circle C_i. The expected number of collisions in C_i before its termination is $\mathrm{E}[C] < 6$ and $C < 7 \log n$ with probability at least $1 - \mathrm{o}\big(n^{-2}\big)$.*

Proof. Only channel Γ_1 is considered in this proof. We assume that C_i is not yet terminated and we define the following events

$$\begin{aligned} A &: \text{Exactly one node in } D_i \text{ is sending} \\ X &: \text{More than one node in } C_i \text{ is sending} \\ Y &: \text{At least one node in } C_i \text{ is sending} \\ Z &: \text{Some node in } D_i \setminus C_i \text{ is sending} \end{aligned}$$

For the proof, we consider only rounds in which at least one node in C_i sends. (No new dominators emerge in C_i if no node sends). We want to bound the conditional probability $\mathrm{P}[A \mid Y]$ that exactly one node $v \in C_i$ in D_i is sending. Using $\mathrm{P}[Y \mid X] = 1$ and the fact that Y and Z are independent, we get

$$\begin{aligned} \mathrm{P}[A \mid Y] &= \mathrm{P}\left[\overline{X} \mid Y\right] \cdot \mathrm{P}\left[\overline{Z} \mid Y\right] = (1 - \mathrm{P}[X \mid Y])\,(1 - \mathrm{P}[Z]) \\ &= \left(1 - \frac{\mathrm{P}[X]\,\mathrm{P}[Y \mid X]}{\mathrm{P}[Y]}\right)(1 - \mathrm{P}[Z]) = \left(1 - \frac{\mathrm{P}[X]}{\mathrm{P}[Y]}\right)(1 - \mathrm{P}[Z])\,. \end{aligned} \tag{4}$$

We can now compute bounds for the probabilities $\mathrm{P}[X]$, $\mathrm{P}[Y]$, and $\mathrm{P}[Z]$:

$$\begin{aligned}
\mathrm{P}[X] &= 1 - \prod_{k \in C_i} (1 - p_k) - \sum_{k \in C_i} \left(p_k \prod_{\substack{l \in C_i \\ l \neq k}} (1 - p_l) \right) \\
&\leq 1 - \left(\frac{1}{4}\right)^{\sum_{k \in C_i} p_k} - \sum_{k \in C_i} p_k \cdot \left(\frac{1}{4}\right)^{\sum_{k \in C_i} p_k} \\
&= 1 - \left(1 + \sum_{k \in C_i} p_k\right) \left(\frac{1}{4}\right)^{\sum_{k \in C_i} p_k}
\end{aligned}$$

The first inequality for $\mathrm{P}[X]$ follows from Fact 1 and inequality (4). Using Fact 1, we can bound $\mathrm{P}[Y]$ as $\mathrm{P}[Y] = 1 - \prod_{k \in C_i} (1 - p_k) \geq 1 - (1/e)^{\sum_{k \in C_i} p_k}$. In the proof for Lemma 2, we have already computed a bound for P_0, the probability that no node in $D_i \setminus C_i$ sends. Using this result, we can write $\mathrm{P}[Z]$ as

$$\mathrm{P}[Z] = 1 - \prod_{C_j \in D_i \setminus C_i} \prod_{k \in C_j} (1 - p_k) \underset{\text{Eq. (3)}}{\leq} 1 - \left[\left(\frac{1}{4}\right)^{\frac{3}{2^\beta}} \right]^{18}.$$

Plugging all inequalities into equation (4), we obtain the desired function for $\mathrm{P}[A \mid Y]$. It can be shown that the term $\mathrm{P}[X] / \mathrm{P}[Y]$ is maximized for $\sum_{k \in C_i} p_k = \frac{3}{2^\beta}$ and thus, $\mathrm{P}[A \mid Y] = (1 - \mathrm{P}[X] / \mathrm{P}[Y]) \cdot (1 - \mathrm{P}[Z]) \geq 0.18$.

This shows that whenever a node in C_i sends, C_i terminates with constant probability at least $\mathrm{P}[A \mid Y]$. This allows us to compute the expected number of collisions in C_i before the termination of C_i as a geometric distribution, $\mathrm{E}[C] = \mathrm{P}[A \mid Y]^{-1} \leq 6$. The high probability result can be derived as $\mathrm{P}[C \geq 7 \log n] = (1 - \mathrm{P}[A \mid Y])^{7 \log n} \in \mathrm{O}(n^{-2})$.

So far, we have shown that the number of collisions before the clearance of C_i is constant in expectation. The next lemma shows that the number of *new dominators per collision* is also constant. In a collision, each of the sending nodes may already be dominator. Hence, if we assume that every sending node in a collision is a new dominator, we obtain an upper bound for the true number of new dominators.

Lemma 4. *Let D be the number of nodes in C_i sending in a time-slot and let Φ denote the event of a collision. Given a collision, the expected number of sending nodes is* $\mathrm{E}[D \mid \Phi] \in \mathrm{O}(1)$. *Furthermore,* $\mathrm{P}[D < \log n \mid \Phi] \geq 1 - \mathrm{o}\left(\frac{1}{n^2}\right)$.

Proof. Let m, $m \leq n$, be the number of nodes in C_i and $\mathrm{N} = \{1 \ldots m\}$. D is a random variable denoting the number of sending nodes in C_i in a given time-slot. We define $A_k := \mathrm{P}[D = k]$ as the probability that exactly k nodes send. Let $\binom{\mathrm{N}}{k}$ be the set of all k-subsets of N (subsets of N having exactly k elements).

Defining A'_k as $A'_k := \sum_{Q \in \binom{N}{k}} \prod_{i \in Q} \frac{p_i}{1-p_i}$ we can write A_k as

$$A_k = \sum_{Q \in \binom{N}{k}} \left(\prod_{i \in Q} p_i \cdot \prod_{i \notin Q} (1 - p_i) \right)$$
$$= \left(\sum_{Q \in \binom{N}{k}} \prod_{i \in Q} \frac{p_i}{1 - p_i} \right) \cdot \prod_{i=1}^{m} (1 - p_i) = A'_k \cdot \prod_{i=1}^{m} (1 - p_i). \quad (5)$$

Fact 3 *The following recursive inequality holds between two subsequent A'_k:*

$$A'_k \leq \frac{1}{k} \sum_{i=1}^{m} \frac{p_i}{1 - p_i} \cdot A'_{k-1} \quad , \quad A'_0 = 1.$$

Proof. The probability A_0 that no node sends is $\prod_{i=1}^{m} (1 - p_i)$ and therefore $A'_0 = 1$, which follows directly from equation (5). For general A'_k, we have to group the terms $\prod_{i \in Q} \frac{p_i}{1-p_i}$ in such a way that we can factor out A'_{k-1}:

$$A'_k = \sum_{Q \in \binom{N}{k}} \prod_{j \in Q} \frac{p_j}{1 - p_j} = \frac{1}{k} \sum_{i=1}^{m} \left(\frac{p_i}{1 - p_i} \cdot \sum_{Q \in \binom{N \setminus \{i\}}{k-1}} \prod_{j \in Q} \frac{p_j}{1 - p_j} \right)$$
$$\leq \frac{1}{k} \sum_{i=1}^{m} \left(\frac{p_i}{1 - p_i} \cdot \sum_{Q \in \binom{N}{k-1}} \prod_{j \in Q} \frac{p_j}{1 - p_j} \right)$$
$$= \frac{1}{k} \sum_{i=1}^{m} \frac{p_i}{1 - p_i} \cdot \left(\sum_{Q \in \binom{N}{k-1}} \prod_{j \in Q} \frac{p_j}{1 - p_j} \right) = \frac{1}{k} \sum_{i=1}^{m} \frac{p_i}{1 - p_i} \cdot A'_{k-1}.$$

We now continue the proof of Lemma 4. The conditional expected value $\mathrm{E}[D \mid \Phi]$ is $\mathrm{E}[D \mid \Phi] = \sum_{i=0}^{m} (i \cdot \mathrm{P}[D = i \mid \Phi]) = \sum_{i=2}^{m} B_i$ where B_i is defined as $i \cdot \mathrm{P}[D = i \mid \Phi]$. For $i \geq 2$, the conditional probability reduces to $\mathrm{P}[D = i \mid \Phi] = \mathrm{P}[D = i] / \mathrm{P}[\Phi]$. In the next step, we consider the ratio between two consecutive terms of $\sum_{i=2}^{m} B_i$.

$$\frac{B_{k-1}}{B_k} = \frac{(k-1) \cdot \mathrm{P}[D = k-1 \mid \Phi]}{k \cdot \mathrm{P}[D = k \mid \Phi]} = \frac{(k-1) \cdot \mathrm{P}[D = k-1]}{k \cdot \mathrm{P}[D = k]}$$
$$= \frac{(k-1) \cdot A_{k-1}}{k \cdot A_k} = \frac{(k-1) \cdot A'_{k-1}}{k \cdot A'_k}.$$

It follows from Fact 3, that each term B_k can be upper bounded by

$$B_k = \frac{k A'_k}{(k-1) A'_{k-1}} \cdot B_{k-1} \underset{\text{Fact 3}}{\leq} \frac{k \left(\frac{1}{k} \sum_{i=1}^{m} \frac{p_i}{1-p_i} \cdot A'_{k-1} \right)}{(k-1) A'_{k-1}} \cdot B_{k-1}$$
$$= \frac{1}{k-1} \sum_{i=1}^{m} \frac{p_i}{1 - p_i} \cdot B_{k-1} \leq \frac{2}{k-1} \sum_{i=1}^{m} p_i \cdot B_{k-1}.$$

The last inequality follows from $\forall i : p_i < 1/2$ and $p_i \le 1/2 \Rightarrow \frac{p_i}{1-p_i} \le 2p_i$.

From the definition of B_k, it naturally follows that $B_2 \le 2$. Furthermore, we can bound the sum of sending probabilities $\sum_{i=1}^{m} p_i$ using Lemma 2 to be less than $\frac{3}{2^\beta}$. We can thus sum up over all B_i recursively in order to obtain $\mathrm{E}\left[D \mid \Phi\right]$:

$$\mathrm{E}\left[D \mid \Phi\right] \;=\; \sum_{i=2}^{m} B_i \;\le\; 2 + \sum_{i=3}^{m} \left[\frac{2}{(i-1)!} \left(\frac{6}{2^\beta} \right)^{i-2} \right] \;\le\; 2.11.$$

For the high probability result, we solve the recursion of Fact 3 and obtain $A'_k \;\le\; \frac{1}{k!} \left(\sum_{i=1}^{m} \frac{p_i}{1-p_i} \right)^k$. The probability $P_+ := \mathrm{P}\left[D \ge \log n \mid \Phi\right]$ is

$$\begin{aligned}
P_+ = \sum_{k=\lceil \log m \rceil}^{m} A_k \;&\le\; \sum_{k=\lceil \log m \rceil}^{m} A'_k \;\le\; \sum_{k=\lceil \log m \rceil}^{m} \left[\frac{1}{k!} \cdot \left(\sum_{i=1}^{m} \frac{p_i}{1-p_i} \right)^k \right] \\
&\le \sum_{k=\lceil \log m \rceil}^{m} \left[\frac{1}{k!} \cdot \left(2 \cdot \sum_{i=1}^{m} p_i \right)^k \right] \underset{\text{Lm. 2}}{\le} (m - \lceil \log m \rceil) \cdot \frac{\left(2 \cdot \sum_{i=1}^{m} p_i\right)^{\lceil \log m \rceil}}{\lceil \log m \rceil !} \\
&\le m \cdot \left(2 \cdot \sum_{i=1}^{m} p_i \right)^{\lceil \log m \rceil} \;\le\; m \cdot \left(\frac{6}{2^\beta} \right)^{\lceil \log m \rceil} \;\in\; \mathrm{O}\left(\frac{1}{m^2} \right).
\end{aligned}$$

The last key lemma shows that the expected number of new dominators *after* the termination of circle C_i is also constant.

Lemma 5. *Let A be the number of new dominators after the termination of C_i. Then, $A \in \mathrm{O}(1)$ with high probability.*

Proof. We define B and B_i as the set of dominators in D_i and C_i, respectively. Immediately after the termination of C_i, only one node in C_i remains sending on channel Γ_1, because all others will be decided. Because C and $D \mid \Phi$ are independent variables, it follows from Lemmas 3 and 4 that $|B_i| \le \tau' \log^2 n$ for a small constant τ'. Potentially, all $C_j \in D_i$ are already terminated and therefore, $1 \le |B| \le 19 \cdot \tau \log^2 n$ for $\tau := 19 \cdot \tau'$. We distinguish the two cases $1 \le |B| \le \tau \log n$ and $\tau \log n < |B| \le \tau \log^2 n$ and consider channels Γ_2 and Γ_3 in the first and second case, respectively. We show that in either case, a new node will receive a message on one of the two channels with high probability during the algorithm's waiting period.

First, consider case one, i.e. $1 \le |B| \le \tau \log n$. The probability P_0 that one dominator is sending alone on channel Γ_2 is $P_0 = |B| \cdot q \cdot (1-q)^{|B|-1}$. This is a concave function in $|B|$. For $|B| = 1$, we get $P_0 = q = (2^\beta \cdot \lceil \log n \rceil)^{-1}$ and for $|B| = \tau \log n$, $n \ge 2$, we have

$$\begin{aligned}
P_0 \;&=\; \frac{\tau \log n}{2^\beta \lceil \log n \rceil} \cdot \left(1 - \frac{1}{2^\beta \lceil \log n \rceil} \right)^{\tau \log n - 1} \;\ge\; \frac{\tau}{2^\beta} \cdot \left(1 - \frac{\tau / 2^\beta}{\tau \log n} \right)^{\tau \log n} \\
&\underset{\text{Fact 2}}{\ge} \frac{\tau}{2^\beta} e^{-\frac{\tau}{2^\beta}} \left(1 - \frac{(\tau/2^\beta)^2}{\tau \log n} \right) \underset{(n \ge 2)}{\ge} \frac{\tau}{2^\beta} e^{-\frac{\tau}{2^\beta}} \left(1 - \frac{\tau}{2^{2\beta}} \right) \;\in\; \mathrm{O}(1)\,.
\end{aligned}$$

A newly awakened node in a terminated circle C_i will not send during the first $\delta \cdot \lceil \log \hat{n} \rceil$ rounds. If during this period, the node receives a message from an existing dominator, it will become decided and hence, will not become dominator. The probability that such an already covered node does *not* receive any messages from an existing dominator is bounded by $P_{no} \leq (1 - (2^\beta \cdot \lceil \log n \rceil)^{-1})^{\delta \cdot \lceil \log n \rceil} \leq e^{-\delta/2^\beta} \in \mathrm{O}\big(n^{-7}\big)$. Hence, with high probability, the number of new dominators is bounded by a constant in this case. The analysis for the second case follows along the same lines (using Γ_3 instead of Γ_2) and is omitted.

Finally, we formulate and prove the main theorem.

Theorem 2. *The algorithm computes a correct dominating set in time* $\mathrm{O}\big(log^2 \hat{n}\big)$ *and achieves an approximation ratio of* $\mathrm{O}(1)$ *in expectation.*

Proof. Correctness and running time follow from Theorem 1. For the approximation ratio, consider a circle C_i. The expected number of dominators in C_i before the termination of C_i is $\mathrm{E}[D] = \mathrm{E}[C] \cdot \mathrm{E}[D \mid \Phi] \in \mathrm{O}(1)$ by Lemmas 3 and 4. By Lemma 5, the number of dominators emerging after the termination of C_i is also constant. The Theorem follows from the fact that the optimal solution must choose at least one dominator in D_i.

References

1. D. J. Baker and A. Ephremides. The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm. *IEEE Trans. Communications*, COM-29(11):1694–1701, 1981.
2. R. Bar-Yehuda, O. Goldreich, and A. Itai. On the Time-Complexity of broadcast in radio networks: an exponential gap between determinism randomization. In *Proc. 6^{th} Symposium on Principles of Distributed Computing (PODC)*, pages 98–108. ACM Press, 1987.
3. U. Feige. A Threshold of ln n for Approximating Set Cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.
4. J. Gao, L. Guibas, J. Hershberger, L. Zhang, and A. Zhu. Discrete Mobile Centers. In *Proc. 17^{th} Symposium on Computational Geometry (SCG)*, pages 188–196. ACM Press, 2001.
5. L. Gasieniec, A. Pelc, and D. Peleg. The wakeup problem in synchronous broadcast systems (extended abstract). In *Proc. 19^{th} Symposium on Principles of Distributed Computing (PODC)*, pages 113–121. ACM Press, 2000.
6. M. Gerla and J. Tsai. Multicluster, mobile, multimedia radio network. *ACM/Baltzer Journal of Wireless Networks*, 1(3):255–265, 1995.
7. L. Jia, R. Rajaraman, and R. Suel. An Efficient Distributed Algorithm for Constructing Small Dominating Sets. In *Proc. of the 20^{th} ACM Symposium on Principles of Distributed Computing (PODC)*, pages 33–42, 2001.
8. T. Jurdzinski and G. Stachowiak. Probabilistic Algorithms for the Wakeup Problem in Single-Hop Radio Networks. In *Proc. 13^{th} Int. Symposium on Algorithms and Computation (ISAAC)*, volume 2518 of *Lecture Notes in Computer Science*, pages 535–549, 2002.
9. F. Kuhn and R. Wattenhofer. Constant-Time Distributed Dominating Set Approximation. In *Proc. 22^{nd} Symp. on Principles of Distributed Computing (PODC)*, pages 25–32, 2003.
10. S. Kutten and D. Peleg. Fast Distributed Construction of Small k-Dominating Sets and Applications. *Journal of Algorithms*, 28:40–66, 1998.
11. L. G. Roberts. Aloha Packet System with and without Slots and Capture. *ACM SIGCOMM, Computer Communication Review*, 5(2):28–42, 1975.
12. P. Wan, K. Alzoubi, and O. Frieder. Distributed construction of connected dominating set in wireless ad hoc networks. In *Proceedings of INFOCOM*, 2002.