

# Local Strategies for connecting stations by small robotic networks\*

Friedhelm Meyer auf der Heide and Barbara Schneider

**Abstract** Consider a group of  $m$  stations with fixed positions in the plane and a group of  $n$  mobile robots, called relays, aiming at building a communication network between the stations consisting of as few relays as possible. We present two strategies for dimensionless, identical (anonymous), oblivious and disoriented relays with limited viewing radius for constructing such a network. These strategies resemble natural strategies of swarms for maintaining formations. A relay does not communicate with others, its decision - whether to remove itself from the system, or where to move - consists only of the relative positions of its neighbors within its viewing radius. We provide a theoretical analysis of worst-case scenarios and upper and lower bounds for the number of relays used by the strategies. In addition, we show some preliminary experimental results.

**Key words:** Mobile robots; self-organization; distributed computing; cooperation; ad-hoc networks

## 1 Introduction

In our research we investigate the construction of a communication network connecting stations in a planar terrain without obstacles. Mobile relays are used to route messages between the stations. Both stations and relays have a restricted viewing

---

Friedhelm Meyer auf der Heide  
Heinz Nixdorf Institute and Computer Science Department, University of Paderborn e-mail: fmadh@upb.de

Barbara Schneider  
Computer Science Department, University of Paderborn e-mail: barbaras@upb.de

\* Partially supported by the European project "Foundations of adaptive networked societies of tiny artifacts (FRONTS)" within the 7th Framework programme, and the DFG-project "Smart Teams" within the SPP 1183 "Organic Computing"

radius and thus are connected via their so-called unit disk graph. They are able to sense the positions of all nodes within this distance. Starting with a large group of relays with positions somewhere in the plane and a group of stations with fixed positions, our goal is to develop local strategies for the movement and removal of the relays that globally result in a network of relays connecting the stations using as few relays as possible. We assume that the initial unit disk graph of the stations and relays is connected and our strategies must keep it connected. A globally optimal solution would place the relays on the edges of a Minimum Steiner Tree connecting the stations.

In Section 2 we present two local, distributed strategies for the movement and removal of the relays. Both versions are very intuitive and do not use any communication, the relays perform their movements using only the sensed positions of the relays and stations within the communication distance. This approach is defined in [1] and called “Interaction via sensing”. In Section 3 and 4 we analyze and compare the performance of the strategies concerning the number of relays used. The theoretical results provide a worst-case analysis, the experimental results are taken from a simulation and we present sample outcomes that show promising results.

### ***1.1 Related work***

Generally, our work belongs to the field of distributed algorithms for robot swarms, where groups of small, cheap robots are used to perform a variety of tasks. Some of these tasks are the formation of geometric patterns, gathering at and convergence to a single point, searching and partitioning the group of robots. A strategy for gathering which is closely related to ours is analyzed in [2]. The setting here also consists of robots which move synchronously, are oblivious and anonymous and have a limited viewing radius. Also, like in our strategies, the calculation of the next position of a robot is based on the center of the smallest enclosing circle around its neighbors. The main difference to ours is the goal of the strategy: there are no fixed stations and the robots gather in one point. Moreover, no robots are deleted and the avoidance of connection loss is different. In [2], the robots are restricted to a movement distance of at most 1 which must be calculated each round by each robot based on its neighborhood.

Another related scenario was presented in [3], where the explorer and the base station in the static setting correspond to our stations. In addition to only using two stations, a communication chain already exists such that every relay knows its predecessor and successor. Also in that scenario, removal of relays is not considered. The only strategy with removals is presented in [4]. It converges very fast, but also only holds for a chain of relays connecting two stations. In addition, the relays need a sense of direction, the chain has to be consistently directed from one station to the other.

## 1.2 The model

In the following, we will refer to both relays and stations together as nodes. Using a communication distance of 1, the unit disk graph of the stations and relays is a graph where vertices correspond to stations or relays and edges exist between two vertices if and only if the two nodes are within communication distance of each other. As stated in Section 1, we assume that the unit disk graph is connected at the beginning, and our goal is to transform the initial unit disk graph into one which still connects all stations and uses as few relays as possible. Note that an optimal selection would place the relays on the edges of a Minimum Steiner Tree connecting the stations, so that neighboring nodes have distance 1. It is assumed that the measurement of the position of neighbors is exact.

To simplify the analysis, we use a synchronous time model with discrete time steps. In each step, nodes are able to sense their environment within the communication distance of 1, to compute their new positions and to move toward this new position up to a distance of 1. This model is called “LCM-Model” (Look-Compute-Move Model) in [5]. We furthermore denote the position of robot  $r$  at the beginning of time step  $t$  by  $p_r^t$ , and the set of neighbors of robot  $r$  at the beginning of time step  $t$ , i.e. all nodes within communication distance 1, by  $N^t(r)$ .

We assume that a relay  $i$  can be deleted from the system if it recognizes that it is not needed. The exact notions of “not needed” are part of the definition of our strategies. (A deleted relay may be assumed to return to some base camp, in order to be used for another task.)

For our analysis of the number of relays finally needed for the communication network, we need the technical assumption that the viewing radius is smaller than 1, i.e. we assume that the disks defining the unit disk graph are open sets. In this case, the initial unit disk graph has the property that connected nodes are in distance of at most  $1 - \delta$ , for some  $\delta > 0$ . We refer to  $\delta$  as the *slackness* of the initial configuration. In case we allow closed disks, we say that there is no slackness.

## 2 The strategies

The following GO-TO-THE-CENTER strategy is executed sequentially, i.e. every time step one relay performs the strategy. The order in which the relays act is given from the beginning and does not change. The time from the beginning of a move of a relay  $r$  to the beginning of its next move is called a *run*. (We note here that our results also hold if we assume a parallel execution of the actions of relay, as long as no relays which are neighbors move in parallel. We have implemented a randomized such strategy.) If relay  $i$  performs the strategy in time step  $t$ , it first observes the exact positions of all its neighbors. It then computes its new position as *the center of the smallest enclosing circle around all positions of its neighbors within its viewing radius*. This center is equivalent to the point that minimizes the

maximum distance to the nodes in  $N^t(i)$ . If there already is a node  $j$  at this position, relay  $i$  deletes itself.

The EXT-GO-TO-THE-CENTER strategy is an extension of the GO-TO-THE-CENTER strategy with an additional deletion rule. Here, before a relay  $i$  moves, it checks if the subgraph of the unit disk graph induced by  $N^t(i)$  is connected. If this is true,  $i$  deletes itself.

The following lemma shows that the unit disk graph stays connected in both strategies.

**Lemma 1.** *For both strategies, the following holds: If the unit disk graph is connected before time step  $t$ , it is also connected at the end of time step  $t$ .*

*Proof.* Let relay  $i$  and  $j$  be neighbors at the beginning of time step  $t$ . At most one of the two relays can change its position in time step  $t$ . If none of them moves, they are obviously neighbors at the end of time step  $t$ . Let  $i$  therefore be the relay performing one of the two strategies. The point computed by  $i$  minimizes the maximum distance to its neighbors, therefore the maximum distance between  $i$  and its neighbors cannot increase:

$$d_{t+1}(i, j) \leq \max_{k \in N^t(i)} \{d_t(i, k)\} \leq 1$$

If  $i$  is not deleted,  $i$  and  $j$  therefore remain within communication distance and the unit disk graph remains connected. If  $i$  is deleted, there are two possibilities:

(1) There is another relay  $r$  at position  $p_i^{t+1}$  with  $N^{t+1}(i) = N^{t+1}(r)$ .  $i$  and  $r$  therefore have the same neighbors in the unit disk graph at the beginning of time step  $t + 1$  and all edges adjacent to  $i$  lie in a circle. If  $i$  is deleted, the unit disk graph remains connected.

(2) Relay  $i$  recognizes that its neighborhood is still connected without  $i$ . Since the deletion of  $i$  only affects its neighborhood, the whole graph remains connected.  $\square$

### 3 Theoretical insights

This section deals with the worst-case analysis of our strategies. We first prove that, after finite time, the unit disk graph does not change any more, and that the convex hull of all nodes converges to the convex hull of the stations. Then we provide upper and lower bounds for the number of relays for scenarios with two or more stations for both strategies. In the following, we assume that the number of relays in the start configuration is  $n$  and that there exist  $m$  stations. The center of a smallest enclosing circle is defined by two or three points (f. ex. [6]) called basis, we also call the nodes at these positions the basis of a relay  $r$ .

**Theorem 1.** *The number of changes of the unit disk graph is bounded by  $O(n(n+m))$ .*

*Proof.* As soon as two relays  $i$  and  $j$  are neighbors, they stay neighbors until  $i$  or  $j$  is deleted. A change of the unit disk graph therefore is a result of the deletion of a relay or of two relays becoming neighbors. As no relays are added to the system,

the maximum number of relays which can be deleted is  $n$ . Moreover, every relay can only find  $n + m - 2$  new neighbors, resulting in a maximum number of changes of the unit disk graph caused by new neighbors of  $\frac{n}{2}(n + m - 2) = \frac{1}{2}n^2 + \frac{1}{2}nm - n$ . The maximum number of changes of the unit disk graph in general is therefore  $\frac{1}{2}n^2 + \frac{1}{2}nm$ .  $\square$

**Theorem 2.** *All relays will eventually be within  $\varepsilon$ -distance to  $CH^*$ , the convex hull of the stations, for every  $\varepsilon > 0$ .*

The proof follows a similar concept of a proof in [2], where it is shown that the robots gather in one point.

*Proof.*

Let  $CH(t)$  denote the convex hull of all nodes after  $t$  steps. We split the proof in the following two parts. They obviously imply the theorem.

- (a) For every  $t \geq t_0$ ,  $CH(t + 1) \subset CH(t)$ .
- (b) There exists no convex polygon  $CH$  s.th.  $CH^*$  is a proper subset of  $CH$  and  $CH$  is a subset of every  $CH(t)$ .

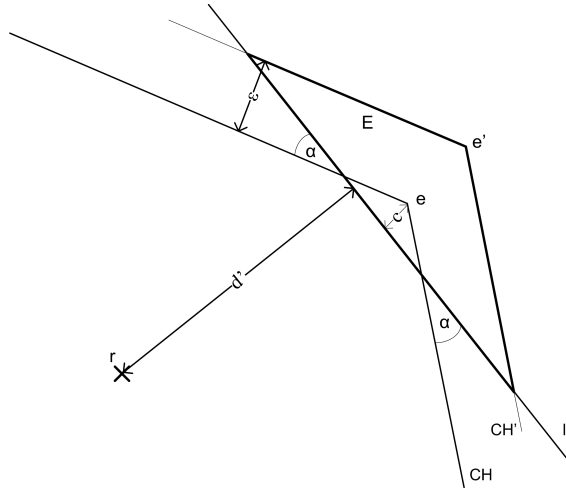
Proof of (a):

Fix some time step  $t + 1$ , in which some node  $i$  moves. This move transforms  $CH(t)$  to  $CH(t + 1)$ . As  $i$ 's new position is contained in the convex hull of its neighbors, it is also contained in  $CH(t)$ , which implies (a).

Proof of (b):

Let  $CH$  be such a convex polygon. Furthermore let  $t_0$  be a time instant such that for all  $t \geq t_0$  all relays are within  $\varepsilon$  distance of  $CH$  for  $\varepsilon > 0$  small enough and such that the unit disk graph does not change any more within finite time. Let  $CH'$  be another convex polygon consisting of edges parallel to those of  $CH$  in distance  $\varepsilon$  outwards from  $CH$ .  $CH$  must have a corner  $e$  which is not defined by a station, since  $CH^*$  is a proper subset of  $CH$ . We call the corresponding corner of  $CH'$   $e'$ . Then there exists a line  $l$  intersecting  $CH$  in distance  $c$  of  $e$  with equal angles to both edges of  $CH$  that intersect in  $e$ , where  $c > 0$  is sufficiently small (compare Figure 1). Let  $E$  be the isosceles triangle formed by  $l$  and the edges of  $CH'$  intersecting in  $e'$ . For every  $t > t_0$ ,  $E$  must contain a set of relays  $R_E$  which can change in every time step, since otherwise the convex hull of all nodes would be a subset of  $CH$  (contradiction to the assumption). As the unit disk graph of the stations and relays is connected, at least one relay in  $R_E$  lies within communication distance to a relay  $r$  outside of  $E$ . W.l.o.g. let  $r$  be the relay with maximum distance to  $l$  which still has a neighbor in  $E$ . By choosing  $\varepsilon$  and  $c$  appropriately, it can be guaranteed that the distance  $d'$  from  $r$  to  $l$  is large enough such that in the run directly after  $r$ 's move all relays in  $R_E$  within communication distance of  $r$  leave  $E$ . Let  $r_i$  be the first relay leaving  $E$  in this run. Then there must still be relays left in  $E$  after  $r_i$ 's move and, since these relays are  $r_i$ 's neighbors, the new position of  $r_i$  must be closer to  $l$  than the position of  $r$ . With  $\varepsilon$  and  $c$  small enough, the whole triangle  $E$  now lies within communication distance of  $r_i$ . Moreover, the distance of  $r_i$  to  $l$  is still large enough so that the new positions of all neighbors of  $r_i$  are outside of  $E$ . This means that in

the following run before  $r_i$ 's next move all neighbors of  $r_i$  leave  $E$  or cannot enter it. Furthermore, every other relay entering  $E$  in this run would become neighbor of  $r_i$  contradicting the assumption. So, on  $r_i$ 's next move, there are no nodes in  $E$ . This is a contradiction to the assumption.  $\square$



**Figure 1** Proof of Theorem 2

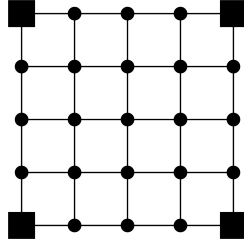
In the following, we prove first upper and lower bounds on the number of relays eventually left over by our strategies. The first bound shows worst case limitations of our strategies.

**Theorem 3.** *Consider four stations forming a square of edge length  $d$ . There are configurations of  $\Theta(d^2)$  relays for both strategies so that no relay will ever be removed. (Note that the minimum Steiner Tree has size  $\Theta(d)$  in this case.)*

The second bound is restricted to the scenario of two stations and shows the asymptotic optimality in this scenario for initial configurations with positive slackness.

**Theorem 4.** *Let an initial configuration with two stations in distance  $d$  and some positive slackness  $\delta$  be given. The number of relays used by EXT-GO-TO-THE-CENTER will eventually be at most  $2d$ . (Note that this is optimal up to a factor of 2.)*

*Proof (of Theorem 3).* If the relays form a grid of distance 0.9 inside the square of stations, there exist edges in the unit disk graph between two adjacent points on the grid (see Figure 2). This is why there exists no relay with a connected neighborhood. Moreover, every relay is positioned in equal distance to all its neighbors in the center of the smallest enclosing circle. Therefore no relay can move or be deleted. The number of relays is then quadratic in  $d$ .



**Figure 2** Illustration for the proof of Theorem 3

*Proof (of Theorem 4).* We first note, that the slackness of a configuration will not decrease anymore after the last of the (finitely many, compare Theorem 1) changes of the unit disk graph. Let the slackness after this last change be  $\delta > 0$ . Now consider a later time step so that, from now on, the relays will always stay within a rectangle around the line connecting the stations with width  $\varepsilon > 0$  small enough so that  $\varepsilon(1 - \delta) < 1$ . Fix an arbitrary relay  $i$ . Because of the choice of  $\varepsilon$ , its neighbors in direction to the first station form a connected subgraph of the unit disk graph, and its neighbors in direction to the other station do so as well. Thus, as the unit disk graph does not change anymore, the closest neighbors of  $i$  in both directions have distance at least 1. This implies the bound  $2d$  for the number of relays.  $\square$

## 4 Experimental results

In this section we will present some preliminary experimental simulation results. In our tests, the results were much better than the worst-case scenarios in Section 3. We have chosen three sample start configurations and present the results for both strategies.

In the first start configuration (Figure 3), the four stations form a square with side length 2.5 and part of the relays are located around the stations in form of a square with side length 5. The remaining of the 53 relays are situated arbitrarily inside or close to the square of relays.

The second start configuration (Figure 5) consists of five stations and 70 relays with the stations situated arbitrarily in the plane and a lot of relays in the convex hull of the stations. The remaining relays are mainly positioned at the left and at the upper right side of the convex hull of the stations and the longest distance between two stations is 6.

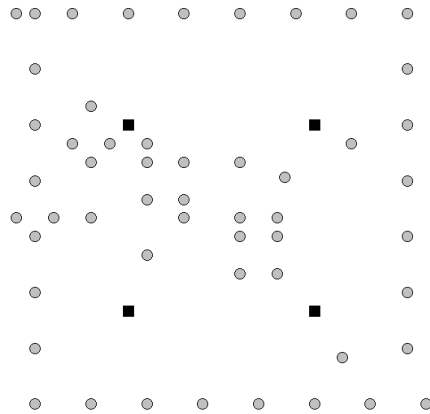
The third start configuration (Figure 7) consists of five stations and 400 relays, all of them positioned uniformly at random in a  $10 \times 10$ -Grid. Because of the big number of relays, the probability for a connected unit disk graph or for connected components which merge during the simulation is high.

These preliminary experiments suggest the following:

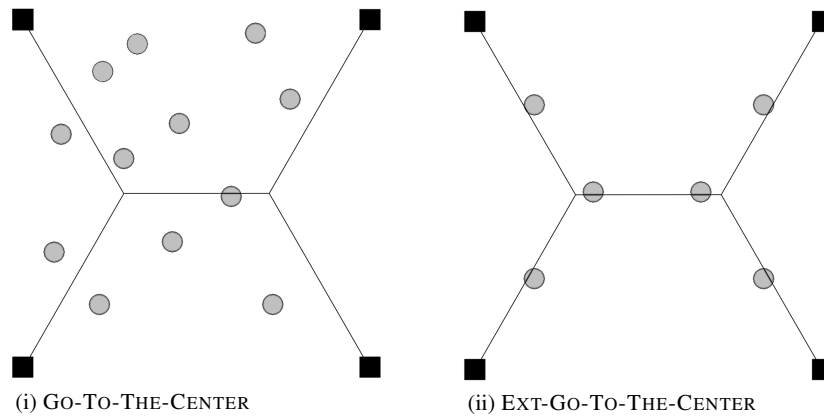
- The set of relays is thinned out considerably.

- The final formation of the relays does not reach the shape of a Minimum Steiner Tree, but comes close.
- Typically, if the formation contains a part that resembles a cycle within the convex hull of the stations, then this cycle cannot be broken by our local strategy.
- EXT-GO-TO-THE-CENTER converges substantially faster than GO-TO-THE-CENTER.

Figure 4 shows the configurations after applying GO-TO-THE-CENTER or EXT-GO-TO-THE-CENTER resp. to the first start configuration, Figure 6 depicts the results for the second and Figure 8 for the third start configuration. Additionally, the figures show the minimal Steiner tree as the globally optimal solution.



**Figure 3** Start configuration for scenario 1



**Figure 4** End configurations for scenario 1



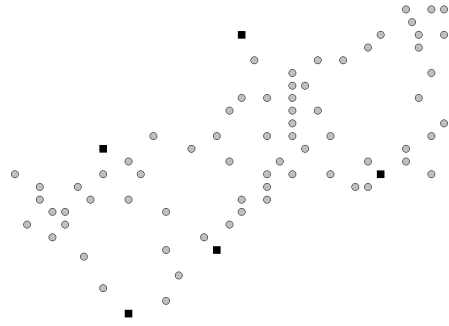


Figure 5 Start configuration scenario 2

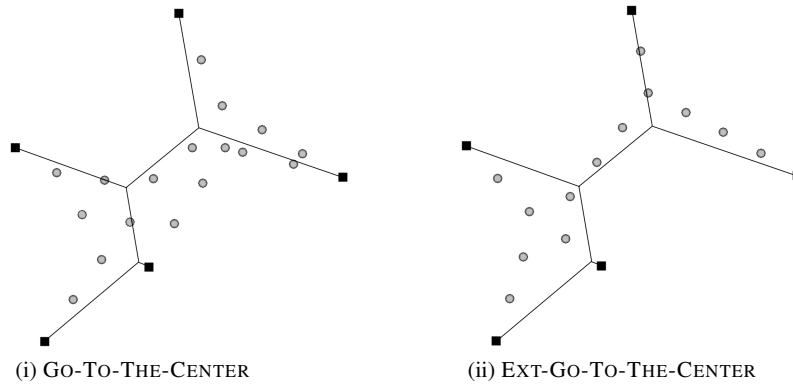


Figure 6 End configurations for scenario 2

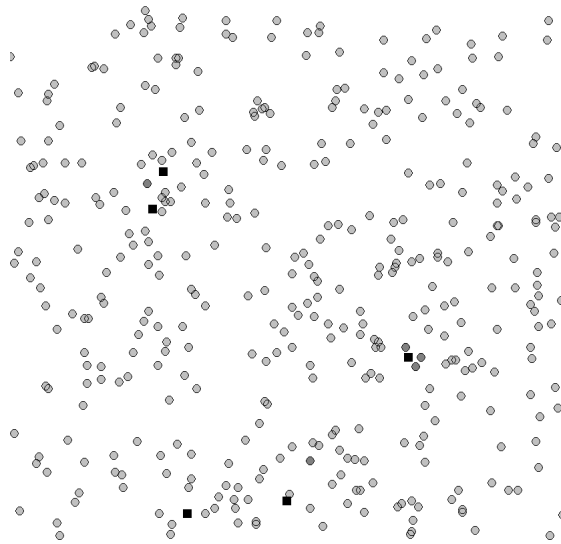
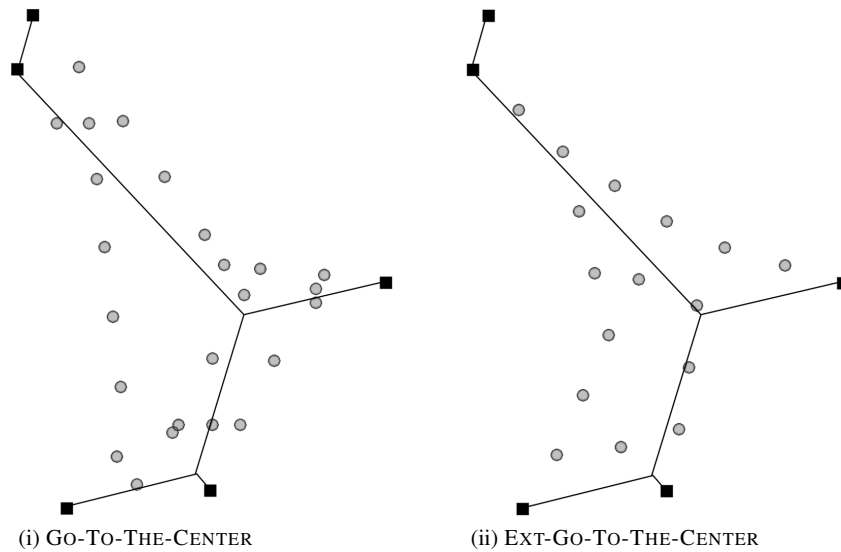


Figure 7 Start configuration scenario 3



**Figure 8** End configurations for scenario 3

## References

1. Y. U. Cao, A. S. Fukunaga, and A. B. Kahng, “Cooperative mobile robotics: Antecedents and directions,” in *Autonomous robots*, vol. 4, pp. 1–23, 1997.
2. H. Ando, I. Suzuki, and M. Yamashita, “Formation and agreement problems for synchronous mobile robots with limited visibility,” in *Proc. IEEE Int. Symp. Intelligent Control*, pp. 453–460, 1995.
3. M. Dynia, J. Kutylowski, P. Lorek, and F. Meyer auf der Heide, “Maintaining communication between an explorer and a base station,” in *Proc. of the 1st IFIP Int. Conf. on Biologically Inspired Cooperative Computing (BICC)*, IFIP, pp. 137–146, Springer-Verlag Berlin, 2006.
4. J. Kutylowski and F. Meyer auf der Heide, “Optimal strategies for maintaining a chain of relays between an explorer and a base camp.” To appear in *Theoretical Computer Science*, 2008.
5. R. Cohen and D. Peleg, “Robot convergence via center-of-gravity algorithms,” in *Proc. of the 11th Int. Colloq. on Structural Information and Communication Complexity (SIROCCO)*, vol. 3104 of *Lecture Notes in Computer Science*, pp. 79–88, 2004.
6. S. Skyum, “A simple algorithm for computing the smallest enclosing circle,” vol. 37, no. 3, pp. 121–125, 1991.