Robot Navigation in a Networked Swarm

F. Ducatelle, G.A. Di Caro, and L.M. Gambardella*

Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), Lugano, Switzerland
{frederick,gianni,luca}@idsia.ch

Abstract. We investigate the use of telecommunications to support the control of a swarm of small mobile robots. The robots need to service events that present themselves in different locations within a confined area. We focus on the task of robot navigation: how can robots of the swarm assist each other to reach event locations. We present two solutions based on the use of routing information set up in a mobile ad hoc network created among the robots. Communication in this network relies on an infrared range and bearing device, which is able to transfer data between two robots as well as to make estimates of the relative distance and angle between them. Using this device, one can relate links in the communication network to relative geographic location information. We then use an ad hoc network routing protocol to dynamically find and maintain paths between a robot and an event location in the communication network and use them to guide the robot to its goal. An important advantage of our approach is that robots can transparently help each other for navigation without having to adapt their own movements, so that they can be involved in independent tasks of their own.

1 Introduction

In this work, we consider a situation where a swarm of robots equipped with wireless communication devices needs to execute tasks in an indoor area. The term "swarm" refers here to a potentially large group of small robots that collaborate using weak coordination mechanisms [11]. The tasks correspond to events that need to be serviced in given locations. Each event can be taken care of by a single robot. A full solution to this problem involves mechanisms for announcing events, for the allocation of robots to events and for guiding robots to event locations. Here we focus on *robot navigation*: how can a robot find an event's location after the event has been advertised. An important aspect of the problem under study is the fact that the robots help each other in their navigation while they are at the same time involved in a task of their own. This is different from most other works, where all robots are involved in solving a single task cooperatively (e.g. collaboratively guide one robot to a destination [17, 14]).

* This work was supported by the SWARMANOID project, funded by the Future and Emerging Technologies programme (IST-FET) of the European Commission under grant IST-022888. The information provided is the sole responsibility of the authors and does not reflect the Commission's opinion. The Commission is not responsible for any use that might be made of data appearing in this publication.

The robots our work is based on are the *foot-bots* which are small mobile robots (about 15 cm wide and 20 cm high) that move around on a combination of tracks and wheels. The foot-bots are part of the *Swarmanoid* project [1], in which also climbing robots (hand-bots) and flying robots (eye-bots) are developed. These other robots have similar communication capabilities to the foot-bots, and will be included in our system in future work. The foot-bots are equipped with three different devices for wireless communication: WiFi, Bluetooth, and an *infrared range and bearing* system (which we will refer to as Ir-RB). While the former two are based on standard technology, the latter is an adaptation of the system presented in [18]. It consist of 26 infrared emitters and 16 receivers, placed around and on top of the foot-bot. Based on the quality of the received signals, it calculates an estimate of the relative bearing and range to other robots using the same system. The maximum range of the system is about 3 meters, and the precision is 20% for range estimates and 30 degrees for bearing estimates. The system also allows line-of-sight communication over the infrared signal with a nominal bandwidth of 40 kbps. The foot-bots are derived from a previous robot called the *s-bot* [13]. Since the foot-bots are currently not yet available our work is based on simulation, whereby the simulator has been derived from an extensively tested s-bot simulator and extended to include new foot-bot features.

We make use of the Ir-RB system to create a *mobile ad hoc network (MANET)* between the robots of the swarm. The Ir-RB system has the advantage that it can couple the reception of each data packet to an estimation of the relative location of its sender, which is information that can be used for navigation. The core idea is to set up a route between a robot that wants to serve an event and the event's location over the MANET. Since each of the robots in the swarm is involved in its own task, this MANET presents frequent and unexpected topology changes. Therefore, we rely on an adaptive routing algorithm, called *AntHocNet*, which is able to find and maintain routes in the face of high network mobility and has been shown to give efficient and robust performance also in large networks and in cluttered environments [8, 9]. The established route is used for robot navigation, whereby we distinguish two modes of operation: in the first the robot *physically follows (robot by robot) the route* formed via the wireless connections, while in the other the route is used to make *estimates of the relative position of the event*, so that the robot can aim to go there independent of the actual route.

Robot navigation has received a lot of attention in robotics research. Early work mainly considers navigation tasks for single robots (e.g., see [5] for an overview of vision based navigation), while in recent years there is an increasing interest in cooperative approaches, whereby communication plays an important role. The simplest form of such cooperative systems is one where a single moving robot is guided by a network of static nodes. These static nodes can be deployed by the robot itself [3], or by an independent process [15]. Localization can be done via GPS [4], or can be based on hop count and/or signal strength measures [15, 20]. Other approaches use a combination of mobile and static nodes. E.g., the authors of [19] let mobile robots move to areas of low connectivity in the network of static nodes. Finally, other systems rely entirely on moving robots, which is the case most similar to our work. In [16] each robot makes a map of its local environment, and communicates it with other robots using multi-hop communication over a MANET. In [14], a swarm intelligence approach is proposed, whereby mobile robots physically form a chain to guide another robot towards a destination goal using visual cues. In [17], the authors present a solution called pheromone robotics, also based on swarm intelligence, whereby robots spread out over an area and indicate the direction to a goal robot using infrared communication. Finally, in [20] mobile robots spread over an area and then choose fixed positions where they serve as beacons that form a communication network and support tasks such as search, localization and navigation.

The system we propose has some important advantages over these previous approaches. First, thanks to the use of the Ir-RB system, robots can get relative positioning information without a central reference system such as GPS. Second, since robots guide each other via wireless communication, they do not need to adapt their movements to help each other (as e.g. is done in approaches where robots serve as beacons to visually guide other robots [14]). Moreover, the use of an adaptive routing algorithm allows to obtain routes even when all robots in the swarm have independent movements. This means that robots can guide each other while they are involved in tasks of their own, which improves the possibilities for parallel task solving. Finally, the possibility to base robot navigation on an estimate of the relative location of the event to be serviced, and independent from the actual MANET route, allows to overcome moments of interrupted network connectivity.

2 Robot routing

The main idea in our approach is to use an ad hoc routing protocol to set up a route between the event and the robot that wants to serve it over the MANET maintained between the robots using their Ir-RB system. We assume that each event is represented by a robot that remains static at the event location and does all the communication for the event. This is a realistic assumption, as the need to perform a task will be identified by one of the robots of the Swarmanoid.

2.1 The MANET routing algorithm

To establish routes in the MANET, we make use of AntHocNet, a MANET routing algorithm based on ideas from Ant Colony Optimization (ACO) [7]. Here we give an high level overview of the algorithm. For more details, see [6, 8].

Under AntHocNet, a node s requiring a route to a destination d sends out a *reactive forward ant*. This is a control packet that has as a task to find a path to d. At any node i in the network, the reactive forward ant can be broadcast (i.e., sent simultaneously to all neighbors of i), or unicast (sent to a specific next hop), depending on whether or not i has routing information available for d. When a node receives multiple copies of the same ant, it forwards only the first one it receives, and discards all subsequent copies. Once the ant reaches d, it is returned to its source node s, following the same path it came over. On its way back, the ant measures the quality of the path and sets up routing information towards the destination. Path quality is measured based on the signal strength of the wireless links along the path. Routing information takes the form of next hop pointers associated to relative goodness values based on the path quality measurements. These goodness values are called pheromone values, in accordance to the ACO inspiration of the algorithm, and are stored in routing tables.

Once the route is set up, the source node s starts a *route maintenance and improvement* process, in order to continuously adapt the existing route to the changes in the dynamic network and find new and better routes when possible. This process is based on two subprocesses: pheromone diffusion and proactive ant sampling. The aim of *pheromone diffusion* is to spread out pheromone information that was placed by the ants. Nodes periodically broadcast messages containing the best pheromone information they have available. Neighboring nodes receiving these messages can then derive new pheromone for themselves (using information bootstrapping, similar to Bellman-Ford updating schemes), and further forward it in their own periodic broadcasts. This way, a field of diffused pheromone is set up that points out possible routes towards the destination. However, since this information is based on the use of periodic (low-frequency) broadcast messages, it can temporarily contain erroneous information. This is where the second process, *proactive ant sampling*, comes in. At constant intervals, node s sends out *proactive forward ants*. Like reactive forward ants, these are control packets that try to find a route towards the destination. They follow the pheromone spread through the diffusion process. When they reach the destination, they travel back to the source setting up a route indicated by pheromone. This way, they update and validate the information found through the pheromone diffusion process and find new routes in the changing MANET.

In case of a route failure, AntHocNet foresees the use of link failure notification messages. When a node i perceives that a link has failed on an existing route (e.g. through a failed message transmission), it constructs a message indicating the address of the destination is has lost a route to. Then, it broadcasts this message to all of its neighbors. Neighbors receiving this message update their routing information accordingly and if this leads to the loss of a route for them too, they send out their own notification message.

2.2 Network routing and robot navigation

When a robot wants to serve a particular event, it uses the AntHocNet routing algorithm to set up a route to the robot indicating the event. Once the route is set up, we foresee two possible ways of using it. The first is that the robot physically follows the route in the network hop by hop. The other is that the route serves to gather information which is used to make an estimate of the relative location of the event, and the robot goes directly there.

Following the route. Following the route set up in the MANET is relatively straightforward. Since the MANET is formed using the Ir-RB system, all robots get continuous measurements of the relative distance and angle to each of their neighbors in the network. There is some error on these measurements: 20% for the distance and 30 degrees for the angle. Therefore, each robot aggregates the received measurements in a moving average, as shown in Equation 1.

$$\begin{aligned}\hat{d}_i^j(t) &= \gamma \hat{d}_i^j(t-1) + (1-\gamma) d_i^j(t)\\ \hat{\alpha}_i^j(t) &= \gamma \hat{\alpha}_i^j(t-1) + (1-\gamma)\alpha_i^j(t)\end{aligned} \tag{1}$$

In this equation, $\hat{d}_i^j(t)$ is robot i's estimate at time t for the distance to neighbor robot j, and $\hat{\alpha}_i^j(t)$ is i's estimate of the angle towards j with respect to its own orientation. $d_i^j(t)$ is the new measurement for the distance received by i at time t, and $\alpha_i^j(t)$ is the new measurement for the angle. γ is a parameter ($\gamma \in [0, 1[$), which defines how quickly the local estimate is adapted to new measurements. This parameter needs to be kept relatively low (we use a value of 0.7 in the experiments), as robots are expected to move a lot and it is therefore not desirable to stick long to old estimates. Using these locally maintained distance and angle values, the robot follows the route by moving towards each hop on the path in turn, until it reaches the destination. The system relies on the continuous availability of a route connecting the robot with the event location. If the route is lost at any time (e.g. due loss of connectivity), the robot remains static and repeatedly tries to establish a new route.

Having the robot physically follow the route has a number of advantages and disadvantages. A first advantage is that it is a simple process. A second advantage is that it provides obstacle-free paths. This is because routes are composed of infrared wireless links, which are only possible if robots are within line of sight from each other. A disadvantage is that the robot can have difficulties following the path when it changes often and abruptly. This can happen when the robots in the MANET move a lot. Also the presence of obstacles can lead to more abrupt changes in the routes. Another disadvantage is that the robot does not know where to move when there is no route available. This leads to low performance in cases of intermittent network connectivity, e.g. when there are few robots around or when obstacles block signals. A final disadvantage is that the path followed by the robot can be substantially longer than the shortest path, especially when the shortest path in the MANET does not correspond to the geographic shortest path (e.g., this can happen when robot density is low [19]).

Making destination location estimations. The other approach is to use the constructed route to give the searching robot an estimate of the relative distance and angle to the event location, so that it can move there directly without following the route. According to the AntHocNet algorithm (see section 2.1), the robot that requests the route periodically sends proactive forward ants towards the route destination (the event location), which are then sent back. We make use of these ants to make the estimates of the destination location: on their way back from the event location to the searching robot, ants gather the locally maintained estimates of the distance $\hat{d}_i^j(t)$ and angle $\hat{\alpha}_i^j(t)$ to each next hop and previous hop (see Equation 1) and combine them to make an estimation of distance $D_i^n(t)$ and angle $A_i^n(t)$ to the event location n. We represent the path followed by the ant as $P = (1, 2, \ldots, n-1, n)$, whereby node 1 is the searching robot and node n is the event location (so the ant travels from n to 1). At any node $i < n$ on this path, $D_i^n(t)$ and $A_i^n(t)$ are calculated according to Equation 2.

$$D_i^n(t) = \begin{cases} \hat{d}_i^n(t), & \text{if } i = n-1, \\ \sqrt[2]{\hat{d}_i^{i+1}(t)^2 + D_{i+1}^n(t)^2 - 2\cos\left(A_{i+1}^n(t) - \hat{\alpha}_i^{i+1}(t)\right)\hat{d}_i^{i+1}(t)D_{i+1}^n(t)}, & \text{if } i < n-1. \end{cases}$$

$$A_i^n(t) = \begin{cases} \hat{\alpha}_i^n(t), & \text{if } i = n-1, \\ A_i^n(t) = \arccos\left[\frac{\hat{d}_i^{i+1}(t)^2 + D_i^n(t)^2 - D_{i+1}^n(t)^2}{2\hat{d}_i^{i+1}(t)D_i^n(t)}\right], & \text{if } i < n-1. \end{cases} \tag{2}$$

Once the robot has received a first estimate of the distance and angle towards the event location, it starts moving towards its goal. As the robot is going, the routing algorithm keeps sending proactive ants regularly, making new estimates available. Having a continuous stream of new estimates is important to overcome errors in previous estimates and to keep an updated view of the event location. Errors in the estimate stem from two main causes. First of all, the event location estimate is based on a composition of local distance and angle estimates along the links of the paths, which each contain some error, and therefore the total estimate has an error that is relative to the number of hops. Hence, at large distances, the event location estimate only offers a rough guideline for the robot's movements, while at smaller distances, the estimate becomes more accurate, so that the robot can eventually zoom in on the event. The second source of errors in the estimates is due to the robot's own movements. As the robot is going, it needs to adapt the estimates of the relative distance and angle to the event location according to its own rotations and displacements, using feedback from an odometry system. This causes the estimate to gradually become less reliable. Therefore, the periodic sending of proactive ants is needed to keep renewing it.

Working with estimates of the relative event location has as an advantage that the robot is not directly dependent on the route itself for its movements. It is sufficient to get a new estimate from time to time. Therefore, there are less problems if the route changes a lot, or if there are periods in which it is impossible to form a route (e.g., due to missing network connectivity or network overload). On the other hand, this approach cannot guarantee an obstacle free path, and it should therefore be combined with an obstacle avoidance mechanism.

3 Experimental results

All tests are done using the Swarmanoid simulator [2], which is derived from simulators developed and extensively tested during the Swarm-bots project [13], the predecessor of Swarmanoid. For each test scenario we execute 30 independent runs. We report the average with 95% confidence interval (using a t-test) of the time needed for the robot to reach the event and of the distance traveled by the robot compared to the straight line distance. We compare three different navigation behaviors: the two communication assisted behaviors of section 2.2 and a sweeping behavior. This last behavior gives a reference of the performance that is possible when no communication is used. In this behavior, the robot

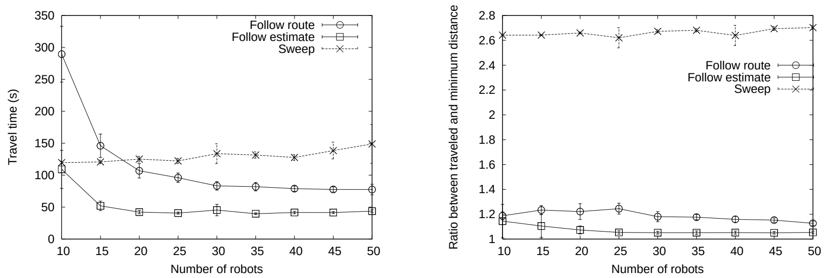

Fig. 1. Results for tests with increasing numbers of robots.

knows its location in the room at all times. It goes to the room corner that is closest to its start location and then starts scanning the room in straight lines parallel to one of the room walls, until it finds the destination. Moving steps in the direction of the other room walls are proportional to the radio range. We use a room of 10×10 m^2, whereby the robot indicating the event and the robot(s) that wants to service the event(s) are located in opposite corners. All other robots move according to random movement patterns, in order to simulate the fact that they are involved in tasks of their own and their movements are independent from the task of guiding the searching robot(s). In particular, they follow the *random waypoint mobility model (RWP)* [12], whereby they choose a random destination in the room, move to that destination and then pause for some time before they choose a new random destination. This model fits well the movements of robots that service sequences of events in the room. All robots are equipped with a minimal obstacle avoidance mechanism.

In addition to the results reported in this paper, we also measured the effect of the ant sending interval and of the accuracy in the odometry measures (see [10] for details). We observed that these two aspects have relatively little impact on performance, except for unrealistically large errors in the odometry measures.

Effect of scaling the number of robots. We investigate the influence of the number of robots on the ability of a searching robot to find an event location. We vary the number of robots from 10 up to 50. The speed of the robots is 0.15 m/s, the pause time of the RWP model is 6 s. The results are shown in Figure 1. We indicate the two communication based behaviors as *Follow route* (the behavior where the robot follows the MANET route) and *Follow estimate* (the behavior where the robot goes straight to the estimated event location), and the sweeping behavior as *Sweep*. As can be seen from the graphs, a lower number of robots makes the task difficult for the communication based behaviors. This is because there is limited network connectivity and the task to establish and maintain a stable route is therefore difficult. This affects especially the *Follow route* behavior, which depends on the constant availability of a route: for this behavior, there is a strong increase in the travel time for the searching robot. For the *Follow estimate* behavior, the increase of the travel time is only visible for the lowest number of robots. Interestingly, the travel distance is not much affected for either of the behaviors. Overall, the *Follow estimate* behavior needs less time and produces shorter paths than the *Follow route* behavior. The *Sweep behavior*

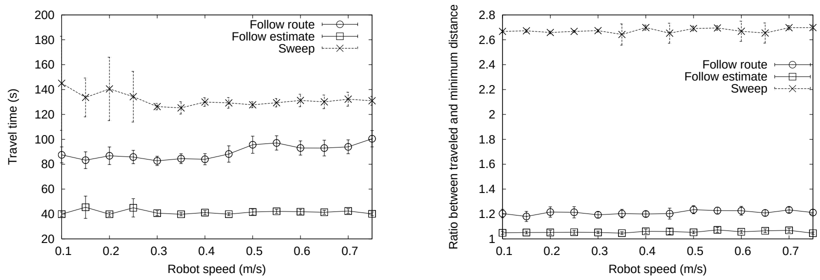

Fig. 2. Results for tests with increasing robot speed.

needs much more time and produces much longer travel distances, especially as the number of robots increases. This indicates the general usefulness of using telecommunications for navigation when a large group of robots is available.

Effect of robot speed. We investigate the influence of the movement speed of the robots on the results. The speed of the searching robot is set to 0.35 m/s. For the other robots, the speed is varied from 0.1 m/s up to 0.75 m/s. The total number of robots is fixed to 30. The results are shown in Figure 2. It is interesting to see that the speed of the robots has little influence on the performance of the system. For the *Follow route* behavior there is a small increase in the robot travel time, but no noticeable effect on the traveled distance. For the *Follow estimate* behavior, there is no noticeable effect for either measure. The higher vulnerability of the *Follow route* behavior is to be expected, as high node mobility leads to higher variability and more frequent disconnections of the MANET route, so that it becomes difficult to follow it hop by hop. The robustness of our approach with respect to robot speed is an important advantage for its deployment.

Effect of the presence of obstacles for a varying number of robots. We place two block obstacles of 1.5×2.5 m^2 in the middle of the room. We vary the number of robots, from 20 up to 60. The results are shown in Figure 3. The presence of obstacles makes it more difficult to establish communication in the MANET. *Follow route* suffers this most: for low numbers of robots, the searching robot requires a large time to find the event, and as the number of robots increases, this time quickly goes down. For the *Follow estimate* behavior, the travel time decreases similarly when going from low to medium numbers of robots, but then increases again for the highest numbers of robots. The *Sweep* behavior shows a similar increase in travel time and also in travel distance when increasing the number of robots. These worse performances for high numbers of robots are due to collisions: since we the number of robots is higher than in the previous tests, and also the obstacles take up some surface of the area, robot collisions become more frequent and influence the results negatively. In general, we can see that the *Follow estimate* behavior gives the best performance.

Effect of the presence of multiple events to service. The number of events to be served is increased from 2 up to 10 in the setup without obstacles. For each event, there is a robot indicating the event and a robot searching it. The total number of robots is 30. This means that as we increase the number of events, we decrease the number of robots following RWP movements and replace them by

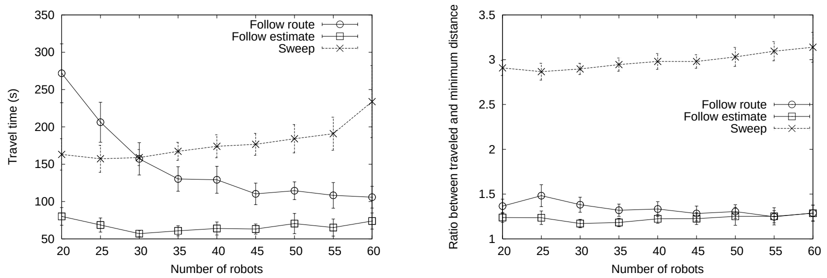

Fig. 3. Results for tests with increasing numbers of robots in the presence of obstacles.

robots indicating and searching events. Consequently, these tests investigate the influence of using different movement patterns. The results are shown in Figure 4. For the *Sweep* behavior there is some increase in the travel time due to robot collisions (since multiple sweeping robots often follow the same trajectories, they can collide), but in general, none of the behaviors is affected very much when multiple events are serviced simultaneously, confirming that our approach allows robots to be involved in independent tasks while helping each other in navigation. As before, the *Follow estimate* behavior gives the best results.

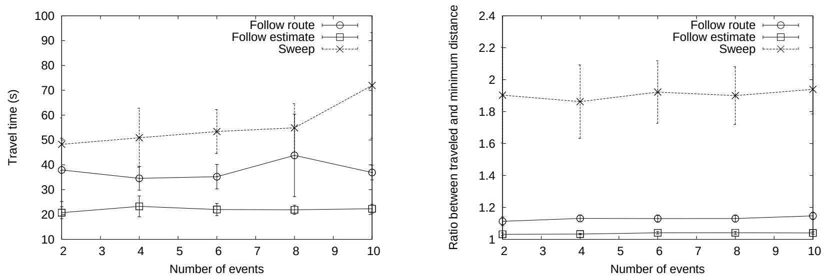

Fig. 4. Results for tests with increasing numbers of events.

4 Conclusions and future work

We have investigated the use of wireless networking to support navigation in a swarm of mobile robots. In our approach, robots use an hoc network routing protocol to discover and maintain paths to event locations in the communication network, and use these paths to find the way to their goals. We propose two algorithms: one in which robots physically follow communication paths, and one in which they use the paths to get estimates of the relative location of their goal. We ran a number of simulation experiments varying the number of robots and their speed in both open space and cluttered environments, and addressed the case of multiple concurrent events. In all cases both algorithms showed robust behavior, with the approach based on making location estimates systematically outperforming the other. In the future, we will extend this work considering also the other types of robots forming the 3D Swarmanoid system. Moreover, we will integrate the navigation function in a more complete system including adaptive task allocation and coordination.

References

1. Swarmanoid: Towards humanoid robotic swarms. `http://www.swarmanoid.org`.
2. Simulator prototype. Internal Deliverable D7 of IST-FET Project *Swarmanoid* funded by the European Commission under the FP6 Framework, 2008.
3. M.A. Batalin and G.S. Sukhatme. Coverage, exploration and delpoyment by a mobile robot and communication network. In *Proceedings of the International Workshop on Information Processing in Sensor Networks*, 2003.
4. P. Corke, R. Peterson, and D. Rus. Networked robots: Flying robot navigation using a sensor net. In *Proceedings of the Eleventh International Symposium of Robotics Research (ISRR)*, October 2003.
5. G.N. Desouza and A.C. Kak. Vision for mobile robot navigation: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):237–267, 2002.
6. G. Di Caro, F. Ducatelle, and L. M. Gambardella. AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks. *European Transactions on Telecommunications (ETT)*, 16(5), 2005.
7. M. Dorigo, G. Di Caro, and L. M. Gambardella. Ant algorithms for distributed discrete optimization. *Artificial Life*, 5(2):137–172, 1999.
8. F. Ducatelle, G. Di Caro, and L. M. Gambardella. Using ant agents to combine reactive and proactive strategies for routing in mobile ad hoc networks. *International Journal of Computational Intelligence and Applications (IJCIA)*, 5(2), 2005.
9. F. Ducatelle, G. Di Caro, and L. M. Gambardella. A study on the use of MANETs in urban environments. Technical Report 01-07, IDSIA, Switzerland, 2007.
10. F. Ducatelle, G.A. Di Caro, and L.M. Gambardella. Robot navigation in a networked swarm. Technical Report 06-08, IDSIA, Switzerland, 2008.
11. A. Farinelli, L. Iocchi, and D. Nardi. Multirobot systems: a classification focused on coordination. *IEEE Trans. on Sys., Man, and Cyb., B*, 34(5):2015–2028, 2004.
12. D. B. Johnson and D. A. Maltz. *Mobile Computing*, chapter Dynamic Source Routing in Ad Hoc Wireless Networks. Kluwer, 1996.
13. F. Mondada, G. Pettinaro, A. Guignard, I. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L.M. Gambardella, and M. Dorigo. SWARM-BOT: a new distributed robotic concept. *Autonomous Robots*, 17(2–3), 2004.
14. S. Nouyan and M. Dorigo. Chain based path formation in swarms of robots. In *Proc. of the 5th Int. Workshop on Ant Colony Optimization (ANTS)*, 2006.
15. K. O'Hara and T. Balch. Pervasive sensor-less networks for cooperative multi-robot tasks. In *Proceedings of the Seventh International Symposium on Distributed Autonomous Robot Systems (DARS-04)*, 2004.
16. S. Park, J. Jung, and S.-L. Kim. Multi-robot path-finding in a random maze with multihop communications. In *Proceedings of the First Workshop on Wireless Multihop Communications in Networked Robotics (WMCNR)*, April 2008.
17. D. Payton, M. Daily, R. Estowski, M. Howard, and C. Lee. Pheromone robotics. *Autonomous Robots*, 11(3), November 2001.
18. J. Pugh and A. Martinoli. Relative localization and communication module for small-scale multi-robot systems. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 188–193, 2006.
19. T.C.H. Sit, Z. Liu, M.H. Ang Jr., and W.K.G. Seah. Multi-robot mobility enhanced hop-count based localization in ad hoc networks. *Robotics and Autonomous Systems*, 55(3):244–252, March 2007.
20. U. Witkowski, M. El-Habbal, S. Herbrechtsmeier, A. Tanoto, J. Penders, L. Alboul, and V. Gazi. Ad-hoc network communication infrastructure for multi-robot systems in disaster scenarios. In *Proceedings of the International Workshop on Robotics for Risky Interventions and Surveillance of the Environment*, 2008.