Portable Sensor Motes as a Distributed Communication Medium for Large Groups of Mobile Robots

Sean Luke
sean@cs.gmu.edu

Katherine Russell
krusselc@gmu.edu

Department of Computer Science
George Mason University
4400 University Drive MSN 4A5
Fairfax, VA 22030 USA

Summary We argue for the use of swarms of distributed portable sensors as a support medium for a large number of autonomous mobile robots. Because of the scaling issues inherent in their multiplicity, and because they may operate in broadcast-denied environments, swarm robot architectures often focus on local and “indirect” communication methods such as breadcrumbs, pheromones, or messages left in the environment. We are interested in how far we can go with these models in real robots. To this end, our research investigates robots capable of deploying, retrieving, moving, and locally communicating with many embedded sensor motes. The mobile agents deploy and optimize the location of the motes, read historic and current sensor data from them, and store useful local information in them for other mobile agents to discover later. We have demonstrated the ability to do robot foraging in environments with significant noise and physical disruption, such as might occur in any deployment of a large sensor network. We have also demonstrated experiments using swarms and sensor motes to collectively build sophisticated, non-trivial swarm behaviors, such as laying out complex shapes using compass/straightedge geometry. In this paper we discuss these results and their limitations, and indicate where we think wireless sensor mote technology can help advance swarm robotics going forward.

Introduction

In this paper we will argue for the value of a swarm of physically embedded, distributed wireless devices as a communication medium for a swarm of mobile robots: that is, a swarm which supports a swarm. The devices of interest to us are *wireless sensor motes*—distributed low-power sensor devices with small on-board computers and radios capable of local communication and ultimately ad-hoc wireless networking. We will illustrate the value of even basic uses of such devices, with nontrivial examples drawn from our previous work in pheromone-based swarm robotics.

Swarms of robots have a great many applications, from driverless road vehicles, to distributed mapping of dangerous areas, to distributed disaster relief and search-and-rescue, to warehouse order fulfillment (for example, the swarms of robots from AmazonRobotics http://amazonrobotics.com). Many of these applications require collaboration beyond simple local coordination of line-of-sight communication, and we believe it is here that swarms of wireless sensor motes may be fruitful.

For robotics applications, wireless sensor motes have a number of very attractive features. First, they are small and can be either on-board the robot, or free-standing in the environment: and because they are small, robots can deploy them or move them as necessary. Second, unlike other embedded elements in the environment (RFID tags, for example), sensor motes are computers: they can store significant information, can perform procedures, and can negotiate in sophisticated ways with robots in the environment or with one another. Third, sensor motes are long-lived and can be placed in the environment by robots to be discovered and used by others much later. Fourth, they can be relatively cheap and easily replaced: it is reasonable to deploy a very large number of sensor motes to serve as a fabric for a smaller swarm of (notionally more expensive) robots.

Note that we have left out the *obvious* features of sensor motes, namely that they have sensors on-board, and they can form ad-hoc networks! While the ability to distribute information in a global manner may ultimately be critical to swarm robotics—and we will speculate on this later in the paper—we first want to focus on their ability to serve, in a scalable manner, as an *indirect communication* mechanism for distributed robot swarms, where robots leave information in the environment for later robots to discover. Indirect communication has emerged as a primary approach to coordinating robot swarms, and our research has studied its capabilities and potential limits. We want to argue that, just considering the indirect communication problem *alone,* swarms of distributed sensor motes are very useful for robotics. And thus augmenting this with their ability to do low-powered, distributed ad-hoc wireless networking, and on-board sensing, could make them indispensable for mobile robotics going forward.

The Communication Problem As the cost of autonomous robots drops precipitously, it becomes possible to have ever-larger swarms of them. For example, our FlockBot ground research robot design (shown later) costs about $500 using custom off-the-shelf parts, and that price could easily be cut to half that with better manufacturing. Many types of autonomous drones can now be built for about this cost as well.

Such robots can perform jobs in a simple parallel or distributed fashion: but many more complex and intricate tasks can be achieved when the robots collaborate or coordinate on them jointly. For example, building a house doesn’t just require large numbers of robots to do separate and independent tasks. Rather certain tasks must be completed before others, some tasks might require heterogeneity in capability amongst the robots, and still other tasks might require multiple robots to work together to achieve them. Such coordination is challenging when the number of robots scales because, in order to coordinate, the robots often must be able to communicate with one another. The scaling problem is straightforward: large numbers of robots will swamp a collective medium such as globally available wireless, and even if not, broadcast methods, or communication with a centralized controller, can quickly overwhelm the listening robots and agents who must deal with very large numbers of irrelevant messages.

As a result swarm research has often focused on local or indirect communication modes in order to combat the scaling problem. An example of local communication is line-of-sight, peer-to-peer communication. The disadvantage of this kind of communication is that, lacking some kind of multi-hop delivery, robots can only easily provide information to local neighbors in the swarm, not distant ones. An alternative, indirect communication, involves storing information in the environment for other robots to discover later. For example, a robot might leave a breadcrumb or signpost in the environment with helpful information for later passersby. Such an approach can potentially reach any robot in the swarm, but is limited to those robots who come across it.

Our indirect communication model uses wireless sensor motes and is directly inspired by indirect communication involving *pheromones* as used by ants or termites. Pheromones have long been used as a model of communication in bio-inspired swarm robotics and artificial life research (Deneubourg et al., 1990; Bonabeau, 1996; Russell, 1999; Payton et al., 2001), but our pheromone model deviates significantly from the biological dogma which permeates such work, and because of this we have been able to make significant strides in what swarm robots can do with indirect communication models. Our approach does not require stigmergic pre-prepared environments, nor special sensors (like chemical sensors), and can be applied to a wide range of environments. Using wireless sensor motes and a pheromone-inspired communication model, we have recently demonstrated collective behaviors well beyond those achieved in the traditional pheromone-based swarm literature: notably construction via compass-and-straight-edge geometry.

In the remainder of the paper, we will discuss previous work in swarm and pheromone models in artificial life and robotics, then introduce our general model, then discuss experiments in deploying it to wireless sensor motes and actual physical robots. We will then conclude with a discussion of our most recent work in collective swarm construction, and argue for future directions in this research which we believe will support the goals set forth for this meeting.

Previous Work

Indirect Communication and Pheromone Models Early work in pheromone models came out of the artificial life community. Much of this literature used grid environments with one pheromone, often to perform foraging tasks where software agents attempted to collect "food" from the environment (Deneubourg et al., 1990; Bonabeau, 1996). Implementing such models with robots, however, requires several modifications. For one, pheromone values must somehow be written to the environment. As the original concept was biologically inspired and there has been work in robotics to produce and sense chemicals, as insects do (Kowadlo and Russell, 2008), some research has followed this model (Russell, 1999; Purnamadjaja and Russell, 2010). Simplifications have also been developed which employ ink (Svennebring and Koenig, 2004), lights (Stewart and Russell, 2006), and even phosphorescent paint (Mayet et al., 2010).

Evaporation and diffusion are common mechanisms in pheromone models, but without chemicals which naturally evaporate and diffuse, other methods must be devised to perform these operations when used in real robot scenarios. The simplest method is obviously to have some sort of global communication medium that can simulate the pheromones as needed, such as a wireless network (Ziparo et al., 2007) or having fixed global communication devices in the environment (Barth, 2003). This unfortunately has the aforementioned scaling issue and may not scale to large numbers of robots. Another option is to use some sort of temporary devices in the environment which can be positioned by the robots themselves with enough abundance that no one device is likely to be overwhelmed. We attempted this with wireless sensor motes (which will be discussed later), but other attempts have used RFID tags (Ziparo et al., 2007), chains of robots (Payton et al., 2001), and even special classes of robots (Ducatelle et al., 2010). If these devices are communicating, then diffusion can potentially be performed through some flooding mechanism. Evaporation, on the other hand, requires only that the devices storing the values have a (somewhat) reliable clock.

Application to Swarm Construction In addition to foraging, multirobot construction is a popular topic in swarm robotics, and typically follows one of a two approaches. First, inert, local features can act as *stigmergic* triggers for a robot to perform some task. For example, if the robot sees the end of a wall, this might "inspire" him to lay down another brick to extend the wall.

Stigmergic methods have been used for ground clearing and site preparation (Parker and Zhang, 2006), building circles around given locations (Pitonakova and Bullock, 2013), and wall construction (Allwright et al., 2014; Stewart and Russell, 2006).

Second, robots may exist in a “smart” environment which can be used to localize robots relative to some global point in the environment or on the structure they are building. For example, a robot building a 3x3 tower of of bricks might count the bricks as it passes by them to determine if it has the right number on a given side, or to determine if it needs to make a 90-degree turn when the third brick is encountered. These models typically require specialized building materials, such as countable bricks or “smart” building materials with embedded information, but quite advanced work has been done using this model, including user-defined 3D structures (Werfel, 2012).

A third, hybrid model, uses markers which can be placed to localize robots relative to some point in the environment. Robots know what tasks they need to perform relative to that location, but have no global context. For example, robots might form a circle some distance from a spot. Such models have produced methods for circle building (Pitonakova and Bullock, 2013), and wall building (Stewart and Russell, 2006).

Swarm Robotic Foraging using Pheromones via Wireless Sensor Motes

Our research work involves robotic construction by large groups of humanoids or differential drive ground vehicles. The figures at right show simple differential-drive swarm robots (our “FlockBots”), each outfitted with a wireless sensor mote, shown suspended above the robot proper. We have 29 FlockBots, though the physical experiments described here use 8–9.

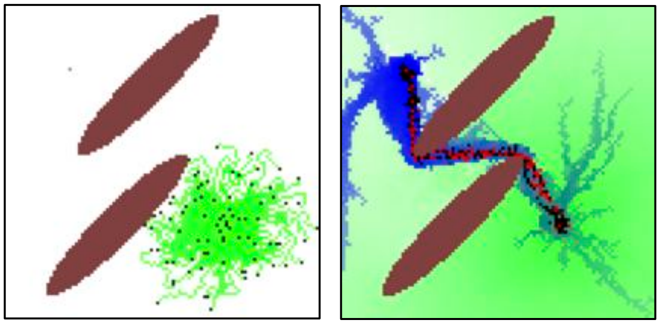
Using its personal sensor mote, a FlockBot may communicate with local free-standing wireless sensor motes in the environment. Each free-standing mote is associated with (and is notionally inside) a can in the environment and a robot is capable of deploying, retrieving, and moving these sensor mote cans as it sees fit. The cans have unique barcodes to make them easy to identify and home in on. Additionally, there is one *nest* at which the robots start at the beginning of an experiment (in the previous figures, the nest can be seen at as the large block at the top left of the image). In our basic model, the robots wander about in a field of wireless sensor motes and perform collective tasks. We assume, for the moment, that the sensor motes do not communicate with one another at all, and similarly the robots do not communicate with one another, but rather a robot can communicate with (read and write data to) any sensor mote in its local range. Furthermore the robots are capable of identifying the set of local sensor motes near to them, and can servo to a sensor mote, either to grab it and move it, or to use it as a waypoint as it wanders the environment.

Each free-standing sensor mote can store various *pheromone values* (positive real-valued numbers), one per pheromone type, which robots can read and write. As needed, motes can also store other auxiliary information, such as locks to enable robots to avoid race conditions when reading and writing these values. On their own, pheromone values stored in sensor motes *evaporate*, meaning that every timestep they decrease by some value (e.g. 0.5%), but they do not *diffuse*, that is, they do not spread into neighboring motes, as the motes do not not communicate. To keep things simple, let us presume that a robot primarily wanders from one sensor mote to a neighboring sensor mote, and only identifies neighboring motes after it has arrived at some mote. Thus we can define a graph with freestanding sensor motes as its nodes, and edges between motes and their neighbors. When not deploying or optimizing motes, each robot is largely traversing this graph.

The pheromone values stored in the motes form gradients among the graph nodes, one gradient surface per pheromone. The robots will traverse the graph by following along a certain pheromone gradient. In addition, as they are traversing the graph, they will revise the values of various (often all) pheromones stored at their current sensor mote so as to update these gradient surfaces. Specifically, to follow a gradient, the robot identifies the local sensor mote whose pheromone value is highest, and then moves to that mote. To update a given pheromone gradient, the robot determines the maximum value of the appropriate pheromone among all the sensor motes in the neighborhood, multiplies it by a slight cut-down factor (perhaps 0.9) and stores this as the pheromone value in the immediate sensor mote. This approach is essentially a collective version of Value Iteration. Some elements are associated with a permanent sensor mote and with a fixed and maximal pheromone value for a pheromone: these serve as peaks in their respective pheromone gradients.

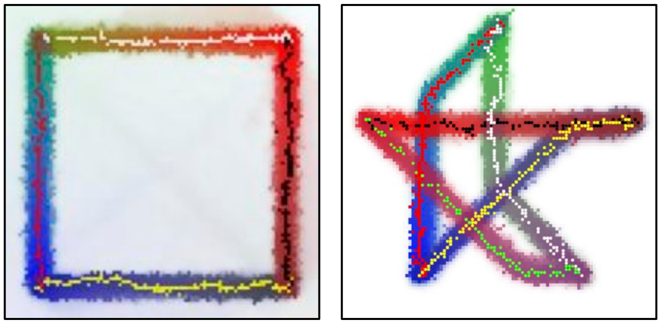
In the *foraging* task, robots fan out to search for a food source, establish and optimize a trail between the food source and the nest, then ferry as much food back to the nest as quickly as possible. There are two pheromones used: *food* and *nest.* The nest is associated with a fixed mote with a maximal *nest* pheromone. The food source (when discovered) is associated with a fixed mote with a maximal *food* pheromone. All other free-standing sensor motes initially have pheromone values of 0. The robots follow simple state machines, and are in either the *foraging* or *ferrying* state. When in the *foraging* state (after starting from or having recently reached the nest), the robot is following the *food* pheromone gradient, if any, while updating the *nest* gradient. Likewise, when the robot has obtained food (it has reached the food), it switches to the *ferrying* state to take food back to the nest, and so follows the *nest* gradient while updating the *food* gradient.

The robots also have some degree of randomness to their actions in order to force exploration: with some probability they may make a random move, or they may enter an *exploration* state whereby they wander randomly away from recently-visited sensor motes.



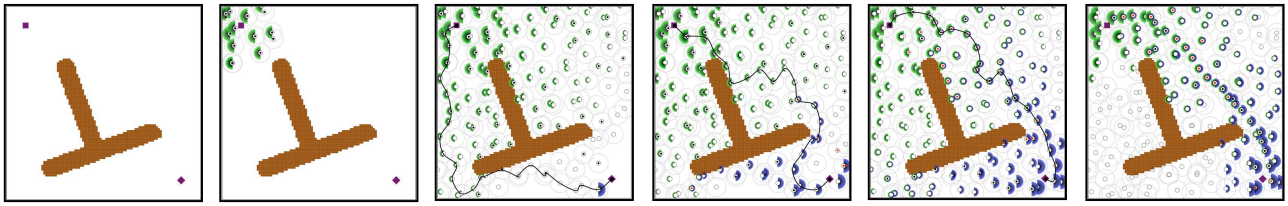
Our earliest simulation paper (Panait and Luke, 2004) assumed that multiple robots wandered about in a 100 x 100 bounded grid environment, and each grid location could hold one or more pheromone values. Implementing this with wireless sensor motes would require 10,000 evenly distributed motes. Multiple robots could be at the same grid location. We showed the agents successfully establishing foraging trails and optimizing them. The figures at right show the robots at the start of simulation coming out of the nest (bottom right in the image) and building the *nest* pheromone (green), then later optimizing a route to the food with both *nest* and *food* (blue) pheromone gradients well established. (Robots in red are ferrying, robots in black are foraging).

Beyond the simple there-and-back-again foraging trails common in the literature, we also demonstrated the robots capable of multi-waypoint tours including potentially self-intersecting paths. The figures at right show four- and five-waypoint trails, which required 8 states (and pheromone types) and 10 states respectively. Note that the five-waypoint trail involved novel self-intersecting paths. We also tested dynamic path optimization, where food sources might move, and introducing new obstacles, forcing the agents to find new routes.



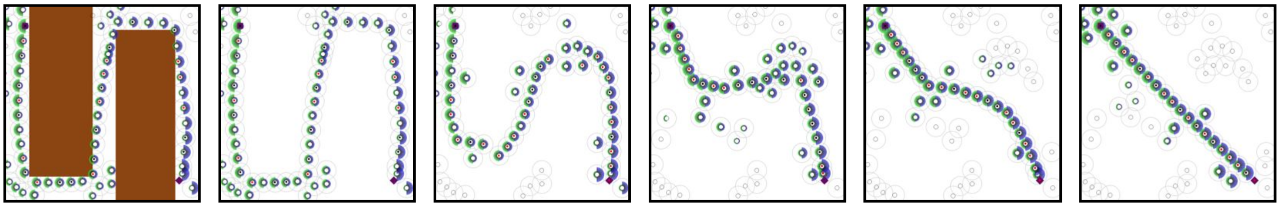
Sensor Mote Mobility

The experiments described above were a reasonable first step, but while pre-deploying a uniform grid of 10,000 fixed sensor motes is *possible,* it may not be plausible for a real mobile robot scenario. Instead, we next moved to allowing the robots to deploy the sensor motes, move them about to optimize their position, and retrieve redundant motes to redeploy elsewhere (Hrolenok et al., 2010). This permitted the robots to embed the pheromone gradients in a sparse and arbitrary (but optimized) graph structure in the environment.



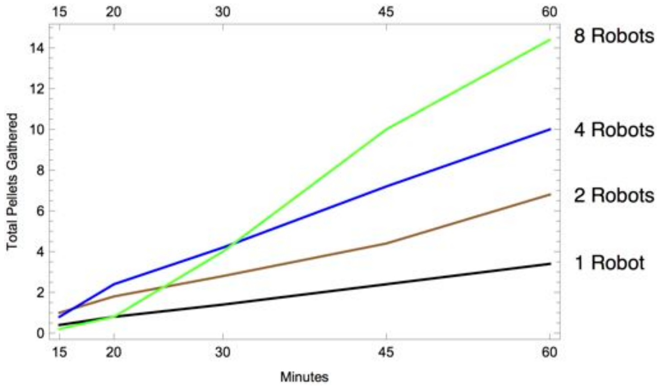
This was again in simulation, using between 60 and 400 motes depending on the trial. The figures above show a typical scenario: the ants leave the nest (top-left in the figures), and deploy sensor motes (green/blue circles, showing the strength of each pheromone value in the mote), then iteratively discover, and ultimately optimize, trails from the food to the nest by moving and retrieving motes.

We also examined dynamic changes in the environment, notably the removal of obstacles which introduced more optimal routes. For example, in the figure sequence below, the robots were first given a chance to establish a path from the nest (top left) to the food (bottom right). After the large obstacles were removed, the robots moved the motes and removed redundant ones in order to optimize the path so as to maximize food gathering.

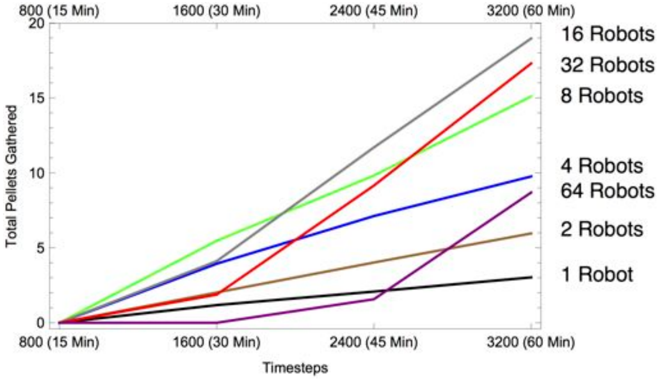


Deployment to Physical Robots

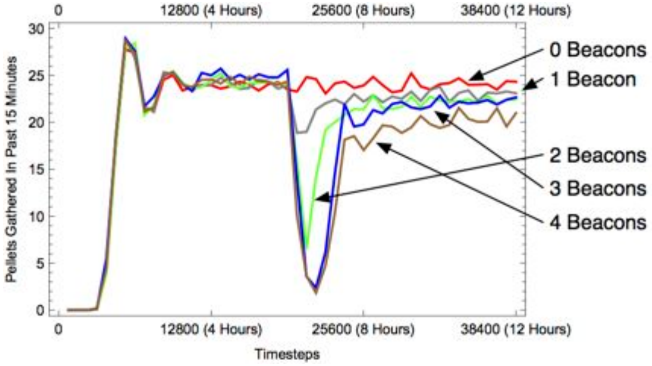
Sparse graphs of mobile sensor motes pave the way for deployment to physical robots. Doing so required revising the algorithm to account for a number of issues, notably the robots taking up physical space and thus visually occluding one another and the sensor motes (recall that our sensor motes are associated with cans wrapped with bar codes). To validate the deployed algorithm, we revised the simulation to introduce physical space, occlusion, sensor and movement noise, and mote failure (Russell et al., 2015).



Once again, our target problem was foraging. At right are two figures showing (top) performance of various numbers of physical robots in our environment and (bottom) the equivalent experiment in simulation. We note that for both the physical robots and the simulated ones performance increased up to 16 robots: but in the simulator, with > 16 robots the performance dropped rapidly. The reason for this is straightforward: crowding. With too many robots, occlusion and in-the-way robots became a major problem.



At right we show one result from our previous paper: recovery from destruction of a large swath of sensor motes (in simulation due to the number of robots required). We would permit the robots to establish and optimize a food trail, and then remove a large diagonal strip of motes across the entire environment. The figure shows well how the robots recover, based on the strip width (in sensor motes, referred to at right as "beacons").



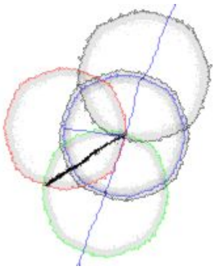
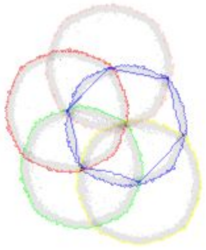
Beyond Foraging: What is Possible? An Unusual Example.

At this point we have demonstrated that pheromones can be used effectively as an indirect communication model to coordinate potentially large swarms of real, physical robots in tasks such as food foraging. Further, we showed that wireless sensor motes deployed in the environment by the robots can be used as the fabric for a pheromone-like communication mechanism, and that this can be done using minimal capabilities of the motes: they only need to talk to local robots, and don't need to establish any kind of ad-hoc network graph.

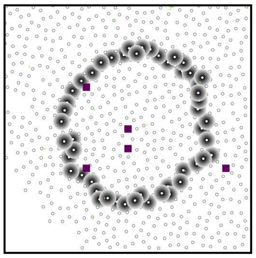
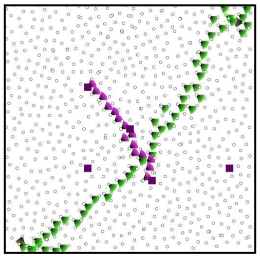
However, foraging is a long-trodden topic: we wondered what else might be possible with indirect communication. Our domain interest is collaborative robotic building construction without using pre-prepared stigmergic building blocks as in (Werfel, 2012) or other domain simplifications. A first step in construction is to lay out sight lines defining the location of walls etc. Such things could reasonably be done using some kind of global shared localization: for example, triangulation on three or more broadcast beacons. However we wondered if it was possible to do this with just indirect communication.

To this end we have recently completed research in applying indirect communication to assist robots in drawing complex geometric shapes in a 2D environment via classic *compass-straightedge construction.* In this technique, derived from Euclid, a geometer is permitted to draw shapes using only an arbitrarily long straightedge and a compass which, when picked up, loses its angle (so it cannot be used to measure distances). Obviously the robots have neither a straightedge nor a compass: they must work together to build the same kinds of constructions. To do this, the robots must be able to do five things: (1) lay out a circle whose center is at some point A and which passes through another point B, (2) lay out an arbitrarily long line which passes through two points A and B, (3) identify the point C at the intersection of two lines, (4) identify and *uniquely distinguish* the two points C and D at the intersection of a line and a circle, or a single point if they are tangent, and (5) identify and *uniquely distinguish* the two points C and D at the intersection of two circles, or a single point if they are tangent.

This is no small feat, particularly given that the robots have no global view of the world. However as we show in an upcoming paper (Russell and Luke, 2016), the robots can in fact do all of these items, and so can collectively achieve nontrivial geometric shapes and tasks. The robots do this using the same pheromone and state machine model as described earlier (with an arbitrary number of pheromones): some pheromones may be set to evaporate, and others may not. The robots also have a single register which can hold a pheromone value, a resettable timer, and the ability to compute relative orientation ("the spot to my left"). The robots use no trigonometry or arithmetic.

The figures at right give two examples (angle bisection and drawing a hexagon, respectively), using the grid-world pheromone model. We think this is one of the most complex tasks achieved in the literature to date using an indirect communication approach.

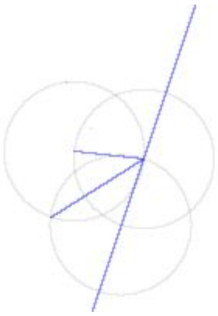



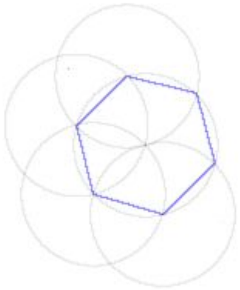
We have also begun work in porting the method to a physical environment with arbitrarily deployable wireless sensor motes and robots. Our first step, as before, is to replicate the result in simulation. Sensor motes are a much coarser and noisier environment, and as can be seen at right (establishing a circle, and performing perpendicular line bisection), our preliminary efforts will require some work to further reduce noise and make the shapes consistent and clean. However our algorithms seem to have transferred successfully with little modification, and we believe we will soon be able to demonstrate them working well here as well.

What's Next: Augmenting the Indirect Communication Model

We think that applications like the compass-straightedge construction example probably sit at the limit of what can be done reasonably with indirect communication alone: because the robots can only communicate via local embedded information, these applications require costly processes with large numbers of robots sufficient to spread information about the environment. Additionally, these models rely on *very* large numbers of sensor motes, which may not be feasible in many situations. For these and other reasons, we think the next step is to violate the pheromone-based canon in various ways, for example:

- Allow a sparser set of sensor motes to *broadcast* over a larger area, and take advantage of relative angle to or distance to motes as a robot sensor feature.
- Allow sensor motes to provide *sensor* information directly to the robots, and consider opportunities in sensor fusion (between the motes and the on-board mobile sensors of the robots).
- Allow sensor motes to form an *ad-hoc network*.





Broadcasting Let us consider the first. In Russell and Luke (2016) we have demonstrated that, were the robots augmented with localization sensors relative to global mote broadcasts, they could perform the same compass-straightedge geometry much more rapidly, much more accurately, and with significantly fewer sensor motes. (Note that the robots *still* wouldn't need to use trigonometry: that is, they wouldn't require triangulation.) To draw a circle, for example, a robot could plant a mote at point A, go out to point B to determine the gradient value from point A at that spot, then follow along the locus of points which have that gradient. The figures at right show the same results using global mote broadcasts: compare to the previous examples of the same. We note however, that while establishing a bearing to a sensor mote is straightforward, establishing *distance* to one or more motes via RSSI or similar methods is not sufficiently accurate, and may need to be improved before such an application becomes a reality.

Sensors and Ad-Hoc Networks Now we consider the ability of sensor motes to form a network amongst themselves, and which the robots deploy and maintain. The robots could employ this network to distribute global coordination events (such as "disaster victim found here!" or "wall-building completed"), to form a hierarchy of command and control amongst themselves, and to spread indirect communication information more rapidly. Were we to also allow the sensor motes to provide their own sensor information, this data could be dispersed and diffused automatically throughout the environment. For example, robot firefighters could create a network of sensor motes in a forest, the sensor motes could generate and diffuse pheromones when they perceived fire hazards, and the robots could follow the pheromones to fight the fires.

The motes might use lower-power, lower-cost, and more general sensors to identify potential threats or targets, and the robots would then move in using both their effectors and more sophisticated on-board sensors to deal with the situations in detail: essentially sensor integration in the form of foveation. Last, the robots could reconfigure the sensor motes so as to optimize their sensor capabilities, enabling self-organization of the sensor swarm.

Sensor motes could do even more for robotic construction: they could coordinate the swarm! In any serious construction task, it isn't enough to assume that a predefined plan can be carried out without a hitch. Robot failures, environment changes, natural disasters,, and any number of other issues might arise during construction. Swarms are designed to be robust in the face of all these (Beni, 2005), but of course pregenerated building plans are not. In this case, some online distributed planner might step in and change the plan to keep the swarm working. The sensor mote network could be used to facilitate just such a hybrid swarm-planner system. The motes could be used to distribute instructions from the planner (or planners), or to provide feedback indicating when various tasks have been completed or events of interest have occurred.

Of course, by permitting global communication via an ad-hoc network, we have reintroduced the significant communications scaling issue that indirect or local communication was meant to avoid. However, we believe that this may not an issue if the global communication is sufficiently slow or sparse. The idea here would be to marry a global communication mode for very occasional communication events with an indirect communication mode for more routine communication needs. We expect that events such as the completion or failure of swarm tasks may not be very common, nor may top-level directives for the robot swarm as a whole to change its tasks. This leads to what we believe will ultimately be the biggest advantage of an ad-hoc network embedded in a robot swarm: it serves as a middle ground between a fully broadcast and fully distributed (local or indirect) communication. Such a network could enable both local and indirect/embedded communication all the time, and global communication when necessary.

Conclusion

We have argued for the value of a swarm-centric sensor network approach in assisting multirobotics applications. Though it would seem that *sensor networks'* value would lie in their *sensor* capabilities and their ability to form ad-hoc *networks* among themselves, in fact much of our past work has completely ignored these features! Rather, we take advantage of their ubiquity, easy deployment, low power, and local communication in order to form a fabric for robotic indirect communication. We have demonstrated that this *alone* is quite valuable can enable swarms of robots to do nontrivial collective tasks. We then ask: what else could such ad-hoc distributed networks enable for robot swarms, when used to their full capacity and promise?

For future work we intend to explore exactly this question. Our ultimate goal is to demonstrate a hybrid of centralized planners and distributed, coordinated swarm behaviors to make possible massively parallel robot swarm construction. A sensor network's ability to serve both as a local or indirect communication mechanism, *and* to form an ad-hoc network graph to serve as a global communication mechanism, may enable both of the communication modes necessary to achieve this. For example, while a robot swarm might use sensor motes for indirect communication in order to collaborate on tasks, a centralized planner/deliberator and executive would use the sensor motes' ad-hoc network to distribute top-level directives to the robot swarm as a whole, or to certain sub-swarms. Critical events ("I found the gold", "task completed", etc.) could be returned to the executive to indicate major changes in the swarm state, triggering new task directives. Similarly, this would provide an easy mechanism for a human in-the-loop to control or otherwise communicate with the swarm as a whole. Relatively little work has focused on hybrid communication or coordination strategies for large swarms of robots, but this will be necessary in order to see them achieve real-world application.

Acknowledgments

The research discussed in this paper was funded in part by NSF grants 0916870 and 1317813.

References

Allwright, Michael et al. 2014. SRoCS: Leveraging Stigmergy on a Multi-Robot Construction Platform for Unknown Environments. *Swarm Intelligence*. Springer International Publishing. 158–169.

Barth, E. J. 2003. A Dynamic Programming Approach to Robotic Swarm Navigation Using Relay Markers. In *American Control Conference.* 5264–5269.

Beni, Gerardo. 2004. From swarm intelligence to swarm robotics. *Swarm Robotics*. Springer Berlin Heidelberg, 1–9.

Bonabeau, Eric. 1996. Marginally Stable Swarms Are Flexible and Efficient. *Journal de Physique I.* 6(2): 309–324.

Deneubourg, J.-L. et al. 1990. The Self-Organizing Exploratory Pattern of the Argentine Ant. *Journal of Insect Behavior.* 3(2): 159–168.

Ducatelle, Frederick et al. 2010. Mobile Stigmergic Markers for Navigation in a Heterogeneous Robotic Swarm. *Swarm Intelligence*. Springer. 456–463.

Hrolenok, Brian et al. 2010. Collaborative Foraging Using Beacons. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems.* 1197–1204.

Kowadlo, Gideon, and R. Andrew Russell. 2008. "Robot Odor Localization: A Taxonomy and Survey." *The International Journal of Robotics Research* 27(8): 869–894.

Mayet, Ralf et al. 2010. Antbots: A Feasible Visual Emulation of Pheromone Trails for Swarm Robots. *Swarm Intelligence*. Springer Berlin Heidelberg. 84–94.

Panait, Liviu and Sean Luke. 2004. A Pheromone-based Utility Model for Collaborative Foraging. In In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems*. 36–43.

Parker, Chris A. C., and Zhang, Hong. 2005. Collective Robotic Site Preparation. *Adaptive Behavior* 14(1): 5–19.

Payton, David W. et al. 2001. Pheromone Robotics. *Autonomous Robots.* Kluwer. 11: 319–324.

Purnamadjaja, Anies Hannawati, and R. Andrew Russell. 2010. "Bi-Directional Pheromone Communication between Robots." *Robotica* 28(1): 69–79. *Cambridge Journals Online*. Web.

Pitonakova, Lenka, and Seth Bullock. 2013. Controlling Ant-Based Construction. In *Proceedings of the Twelfth European Conference on the Synthesis and Simulation of Living Systems* (ALIFE). 151–158.

Russell, Katherine et al. 2015. Swarm Robot Foraging with Wireless Sensor Motes. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*. 287–295.

Russell, Katherine, and Sean Luke. 2016. Ant Geometers. In *15th International Conference on the Synthesis and Simulation of Living Systems* (ALIFE). To Appear.

Russell, R. Andrew. 1999. Ant Trails - an Example for Robots to Follow? *Proceedings of the 1999 IEEE International Conference on Robotics and Automation.* 4: 2698–2703.

Stewart, Robert L., and R. Andrew Russell. 2006. A Distributed Feedback Mechanism to Regulate Wall Construction by a Robotic Swarm. *Adaptive Behavior* 14(1): 21–51.

Svennebring, Jonas, and Sven Koenig. 2004. Building Terrain-Covering Ant Robots: A Feasibility Study. *Autonomous Robots* 16(3): 313–332.

Werfel, Justin. 2012. Collective Construction with Robot Swarms. *Morphogenetic Engineering*. Springer Berlin Heidelberg. 115–140.

Ziparo, V.A. et al. 2007. RFID-Based Exploration for Large Robot Teams. In *IEEE International Conference on Robotics and Automation*. 4606–4613.