

Compound behaviors in pheromone robotics

David Payton, Regina Estkowski, Mike Howard

HRL Laboratories, LLC 3011 Malibu Canyon Road, Malibu CA 90265

{payton, regina, howard}@hrl.com

Abstract: We are pursuing techniques for coordinating the actions of large numbers of small-scale robots to achieve useful large-scale results in surveillance, reconnaissance, hazard detection, and path finding. Using the biologically inspired notion of “virtual pheromone” messaging, we describe how many coordinated activities can be accomplished without centralized control. By virtue of this simple messaging scheme, a robot swarm can become a distributed computing mesh embedded within the environment, while simultaneously acting as a physical embodiment of the user interface. We further describe a set of logical primitives for controlling the flow of virtual pheromone messages throughout the robot swarm. These enable the design of complex group behaviors mediated by messages exchanged between neighboring robots.

Keywords: swarm robotics, cooperative behaviors, distributed control, emergent behavior

1. Introduction

Emerging technologies in micro machining and MEMs hold the promise of creating extremely small robots. Although limited in size and power, such robots can work together in large numbers to conceivably accomplish a wide range of significant tasks [5][7][10][15] including surveillance, reconnaissance, hazard detection, path finding and payload conveyance.

Coordinating and interacting with very large numbers of robots involves issues not encountered when dealing with one or a few robots[2][8][11]. Coordination schemes that require unique identities for each robot, explicit routing of point-to-point communication between robots, or centralized representations of the state of an entire swarm can be overwhelmed when dealing with extremely large numbers. To assure that our multi-robot system is both scaleable to large robot swarms and tolerant to individual robot failures, we focus on techniques that effectively limit each robot’s interactions to its own local neighborhood.

We are inspired by techniques used by ants and termites for communication and coordination [4][9][17]. We implement “virtual pheromones” using simple transceivers mounted on each robot. Like their chemical counterparts, our virtual pheromones facilitate simple communication and emergent coordinated movement with only minimal on-board processing. But virtual pheromones go a step further, transforming a robot swarm into a distributed computation grid embedded in the world. This grid can be used to compute non-local information about the environment such as bottlenecks and shortest paths, in ways that are foreign to insect colonies.

In our approach, we envision a scenario in which a rescue team enters an unfamiliar building after a disaster, and needs to quickly locate any survivors. We imagine multiple tiny robots released at the entrance to the building. Using simple attraction/repulsion behaviors, these robots quickly disperse into the open spaces. Upon detection of a survivor, a robot emits a virtual pheromone message signaling the discovery. This message is propagated locally between robots only along unobstructed paths, producing a gradient as it is propagated. Ultimately, the message makes its way back to the entrance where rescue team members can now follow the pheromone gradient to the survivor. To do this, the robots themselves serve as a distributed display of guideposts leading the way along the shortest unobstructed

path. We are pursuing this vision with 20 small robots specially designed to support our virtual pheromone type communication protocols. In an earlier paper [14] we described our approach. In this paper we briefly review key features of the approach, and then focus on recent developments.

2. Overview of the approach

There are three key features of our approach: *Virtual Pheromones* is a diffusive local-neighborhood interaction mechanism by which the robots communicate and coordinate. *World-Embedded Computation* is a technique for performing many graph-theoretic algorithms in a distributed fashion, without requiring a centralized intermediate representation. *World-Embedded Display* is a method for interfacing to a robot swarm by using the robots as annotations on the real world. These features will be described in the next three subsections.

2.1 *Virtual Pheromones*

Inspired by the chemical markers used by ants and termites for communication and coordination, we developed the notion of a “virtual pheromone,” implemented by messages relayed from robot to robot, with specific features that enable them to travel in decaying wavefronts from the origination point. Virtual pheromones facilitate simple communication and coordination and require little on-board processing. This approach is applicable to future robots with much smaller form factors (e.g., to dust-particle size) and is scaleable to large, heterogeneous groups of robots.

The design of virtual pheromones preserves some of the essential properties of natural pheromones that make them effective in facilitating group organization. (1) Pheromones are locally transmitted without specifying a recipient, obviating the need for unique identities that are impractical in large groups [8]. (2) Pheromone diffusion gradients provide important navigational cues and also encode useful information about barriers in the environment that block pheromone propagation. (3) Pheromones decay over time, reducing obsolete or irrelevant information.

Virtual pheromones are not faithful copies of chemical pheromones, for several practical reasons. For example, natural odors diffuse in patchy plumes whose concentration does not fall off uniformly with distance or time. But virtual pheromones are transmitted at a known intensity, and their signal strength decreases linearly with distance. Therefore, receivers can reliably estimate distances on the basis of signal strength alone. Second, if an originating source for a virtual pheromone moves, the gradient will adjust quickly without the persistence of chemical pheromones. Third, virtual pheromone messages can contain optional data beyond pheromone type, for use in distributed computations.

2.2 *World-Embedded computation*

Traditional approaches to path planning and terrain analysis operate on an internal map of terrain features [12][13] or they perform the steps of sensing, data transmission to a central point, and map generation before the data can be processed. This is especially disadvantageous when the environment is rapidly changing. Pheromone robots (or *pherobots*) require no centralized map, nor do they maintain a detailed history.

Our approach treats pherobots as embedded processing elements in the environment. Each actively senses and computes simultaneously based only on local terrain features and pheromones passed by neighboring robots. Inter-robot communication is via line of sight signaling because it results in a communications grid that embodies mobility costs in its connectivity structure. Rather than trying to

overcome communication loss due to obstacles, we exploit this effect to sense the terrain and determine optimal traversal paths.

In fact, our rules for message propagation provide a distributed version of the wavefront propagation method used in Dijkstra's shortest-path algorithm [6]. Global properties such as shortest routes, blocked routes, and contingency plans are computed in a robust, distributed manner, with each member of the population of simple processors contributing a small piece of the result.

2.3 World-Embedded Display

Pherobots implement an efficient and versatile distributed computer. In some applications the results of the computation are sent back to the user via relayed messages. But the real novelty of the system is when the robot swarm acts as a distributed display embedded in the environment. In effect, each robot becomes a pixel, or an annotation on the immediate environment. The robot's position with respect to the user provides context to interpret the meaning of the transmitted information.

As an example, consider the notional view in Figure 1. Here, the robot swarm has dispersed, and a pheromone gradient has been established. A user, seeking to follow that gradient, needs only to be able to see the local gradient vector at each robot location. One way to read the distributed display is to use an augmented reality (AR) system. AR refers to the visual presentation of information, in geometric registration with true objects seen in the environment. The combination of this world-embedded interface with our world-embedded computation means that the results of complex distributed computations can be mapped directly onto the world with no intermediate representations required.



Figure 1: User's view of a pheromone gradient as seen through an augmented reality display

3. Progress

The rest of the paper will relate the progress we have made toward realizing the vision described in section 2. In particular, we briefly summarize the existing robot hardware and previously reported capabilities, then detail advances in the pheromone control system including command pheromones and controller switching logic, a new implementation of the locomotion control logic, details of some new behaviors, and the latest advances in the user interface.

3.1 The robot platform

We use a swarm of 20 custom designed pherobots (Figure 2). The control system was originally written in Java, using the Teambots [16] environment which also allowed us to simulate algorithms. Java on the PalmVx robot controller, using a small virtual machine called WABA, proved too slow. By porting to C++ and making some other speedups we have reduced cycle time from 3 seconds to under 0.4 seconds.

Atop each robot is a set of eight radially-oriented directional InfraRed (IR) transmitters and receivers (transceivers) as shown in Figure 3. Baffles are installed between transceivers to reduce sensor overlap. Sensing is a side effect of communication; when a robot receives a message it just sent, it is considered a reflection off of an obstacle. Distance to the obstacle is calculated as a function of the received power of the reflected signal. If there is no message to send out of a particular port, an “obstacle ping” is sent to sense obstacles. Robots sense distance and



Figure 2: A Pherobot swarm

direction to other robots by detecting their transmissions. Further details on the logic for message processing are given in section 3.3.

Our choice of this particular robotic platform was a tradeoff between size and robustness. A small platform forces us to confront issues of low power and limited mobility, and indeed we initially considered a much smaller platform. However, the smaller the platform, the more time is spent struggling with practical issues that distract from theoretical issues. Our pherobots provide a reasonable compromise.

3.2 Locomotion and Behavior Primitives

Our robot behaviors, environmental representations, and data structures use only local interactions and information. The primary behavior primitives are based on notions of attraction and repulsion. Repulsion keeps robots from colliding with obstacles and other robots and urges the robot swarm to spread out and maximally cover a space. Robots can also be repelled by certain types of virtual pheromones, so that one robot can serve to block others from certain areas. Attraction keeps robots in communication range with each other and allows them to follow contours of terrain features, or to follow a pheromone message gradient to its source. These elemental behaviors may be combined to produce a variety of emergent group behaviors [1].

For example (Figure 4), one important group behavior is to expand a robot swarm from a tight grouping into a maximal dispersion that maintains nearest-neighbor communications, emulating the way gas particles expand to fill a vacuum. Each robot’s control system generates a repulsion vector to get away from things that are too close, and an attraction vector to get closer to things that are too far away. In between is a neutral zone, to avoid attraction-repulsion oscillations.

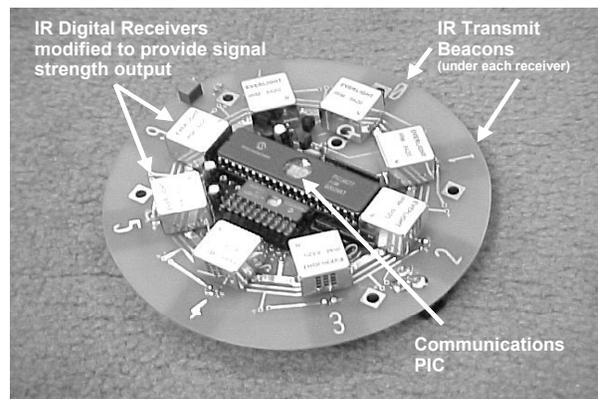


Figure 3: Transceiver for virtual pheromones

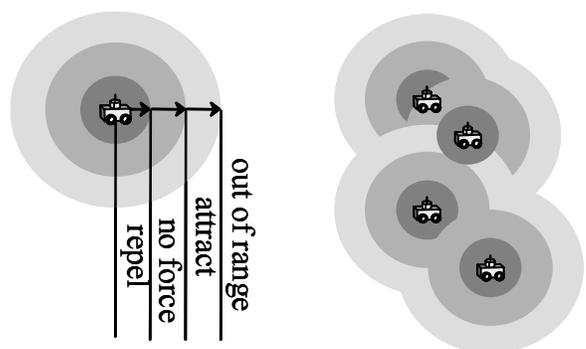


Figure 4: Attraction and repulsion zones implement gas expansion

The necessary complement to expansion is contraction; an effective approach is described in [7].

While standard potential-field methods can be used to compute net attraction or repulsion by summing force vectors between a robot and its local neighbors, it is often more effective to reason about the free space surrounding a robot. To do this, we provide what we call a *bins representation* of free space. The bins representation partitions space surrounding a robot into equally sized pie-like 2D sectors with the robot at the center. These sectors can be made to correspond identically with the sectors formed by the IR sensors, or they can be made smaller or larger. For each sector, we store the range to the nearest object detected within that sector. From this simple representation, we can perform a few basic logical operations that provide us with mobility primitives that would otherwise be difficult to obtain from vector operations alone.

Foremost among these operations are those that look for contiguous blocks of free space around a robot. We can use these operations, for example, to determine if a robot is surrounded, to find the largest open space for a robot to move into, or to locate an empty sector that is close to the robot's current heading. We can also apply thresholding operations such that sectors are considered empty so long as they contain no object within a given distance. We can further exploit this idea by allowing merging of bins representations of different types of objects. This is used in some algorithms to merge robot objects with non-robot objects, to get a unified map for obstacle avoidance. The bins representation is particularly helpful for reasoning about whether a robot is on the edge of a swarm and can move quickly to open space, or if it is in the middle of the swarm and should remain stationary.

3.3 Pheromone Logic

We use multiple pheromone types to direct behaviors and to activate particular control modes. As mentioned in section 3.1, sensing is a side effect of communication. If the sender uses a well-known transmission power (or the power is encoded in the transmission), the receiver can calculate distance as a function of the loss of signal strength. Thus every received message gives information about the distance and direction of an object, whether the object is a robot that sent the message, or just an obstacle that reflected the message. Pheromone messages have several fields: a field-of-view (FOV), a type code, a hop count, and a data field. The FOV is used in algorithms for directional messaging through the swarm. The FOV specifies the width of the beam, while the vector direction determines the center axis. The type code, hop count and data fields make up the actual contents of a virtual pheromone message.

Upon receipt of a virtual pheromone message, a robot converts it into a vector representation corresponding to the direction from which it was received. This message is then processed through logic primitives that result in motion commands and/or re-transmission of a new message. The most central communication logic is that for producing pheromone gradients. This logic decrements the hop-count field of a received message and re-transmits the message in some or all directions. If a robot receives the same type of pheromone from multiple directions, the robot selects the message with the highest hop-count value for re-transmission. This is

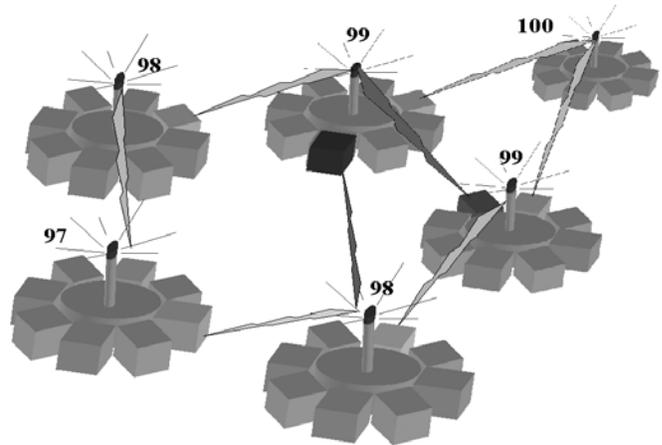


Figure 5: Virtual pheromones are relayed with a lower hop-count by each subsequent robot.

considered the “best” message for the given pheromone type. Any pheromone messages of the same type received with hop-count values equal to or less than the hop-count already transmitted are ignored. These propagation rules are illustrated in Figure 5. The net result is a pheromone gradient expressed as uniformly descending hop counts from the originating source.

There are many other types of pheromone logic that we need to construct compound behaviors. For this purpose, we have designed a set of primitives that can be assembled in a variety of different ways. Table 1 shows some pheromone message primitives we have developed, sorted into three basic types of interaction between the control system and the messaging system. There are requests for pheromone message data, repeating processes such as continual or conditional message forwarding, and commands to transmit an individual pheromone message.

Pheromone Data Request Primitives	Have I sensed type X? Give me all of type X Give me “best” of type X Give me “best” of type X in each bin
Pheromone Message-Passing Primitives	Forward X when Y is true/false Forward X, incrementing (or decrementing) hop count by P Forward X, setting FOV and pheromone type Forward last X for K time cycles Forward X for K cycles then nothing for T cycles Forward X relative to received direction Forward X or Y depending on which has the max or min hop-counts. Forward X or Y, alternating between them.
Pheromone Sending Primitives	Initiate Sending X (once, forever, for time T) Send X with priority P Send X in directions (N S E W...)

Table 1: Pheromone Messaging Primitives

We created most of the above functionality in a set of classes designed for pheromone messaging, derived from Teambots Clay. Like normal clay classes, the “Pheromone Clay” classes may be linked together to form a logical data flow path from received messages to output messages. Along this path, operations such as changing hop-counts, or changing field-of-view may be performed in successive stages. The internal vector representation of pheromones also allows them to easily be integrated into movement behaviors through the standard vector summation primitives in Clay.

A *command pheromone* is a special message that specifies which control system a robot should run. Each robot has a master control system that listens for command pheromones, and upon receipt, switches to the subordinate control system specified by the data field. The data field of the command pheromone also specifies whether the master control system should relay the command pheromone to other robots. Using this option, we can send command pheromones that change only the control state of a single designated robot, or that change the control state of the entire swarm.



Figure 7: Augmented reality mast projects information from the robots to a user’s head-mounted display.

3.4 User Interface



Figure 6: Coded infrared signals are decoded into arrows for overlay onto a real scene. These are visible via the augmented reality head-mounted display system with camera.

The world-embedded display, depicted in Figure 1, is created by using an augmented reality (AR) system [3] combined with specially designed beacons mounted on each robot. The head-mounted display (HMD) worn by the user detects specially coded infrared signals emitted by each robot and displays the information as a graphical overlay on the user's field of view. The infrared signals are detected by a monochrome camera that is mounted atop a Virtual-IO head-mounted display as shown in Figure 6. The camera, filtered to receive only infrared light, receives the infrared signals as blinking dots on a black background. These blinking dots are then tracked and are decoded into arrows that are displayed on the HMD. Proper alignment of the camera with the user's display allows the arrows to be superimposed directly over the robot's positions without any explicit knowledge of the robot or user location. The real-time tracking of

the blinking dots allows the decoded arrows to appear to float over the robots even as the user's head moves. Figure 6 shows the view presented to a user looking at two robots.

The infrared beacons (Figure 7) on each robot are designed such that a robot can display an arrow at a desired direction regardless of the user's orientation relative to the robot. Consider, for example, a user looking at a robot from the left side and seeing an arrow pointing to the left. Another user, looking at that same robot from the right side should see an arrow pointing to the right. We achieve this desired result by having the beacons on each robot transmit different coded messages in each of eight different directions. Depending on the user's orientation with respect to the robot, they will receive a different coded message. Therefore, they will see a different arrow in their HMD, but it will always appear to be pointing in the same direction relative to the robot.

3.5 Compound Behaviors

Complex coordinated robot behaviors can be composed from locomotion and pheromone primitives. In this section we will briefly describe three representative behaviors that we have demonstrated: Gradient Following, Go Hide, and Cooperative Sensing.

3.5.1 Gradient Following

Gradient following is one of the most basic of our compound behaviors. It is also a central element to many of our more complex behaviors. The purpose of gradient following is to enable a dispersed robot swarm to guide one or more robots to a particular area or object of interest. The process begins when a source robot in a distributed swarm initiates a pheromone gradient as described in Section 3.3. Another robot, the follower, is designated to be attracted to this pheromone. The follower travels

towards the direction of the message with the highest hop count received, avoiding any other robots along the way. As the follower approaches the sender of that message, it comes in range of that robot's predecessor in the gradient, which has a higher hop count. The predecessor then becomes the new destination of the follower. In this way, the robot climbs up the gradient until it reaches the originating source robot.

3.5.2 Go Hide

At times we want the entire swarm to run for cover to avoid detection and/or injury. Currently "cover" is defined as the nearest obstacle, usually a wall. When commanded to hide, robots that are able to detect a wall, move towards it, avoiding other robots. They space themselves along the wall and park, and emit a "wall" pheromone to attract robots that may not have detected a wall. Any robot that does not detect a wall travels in the direction of the closest received "wall" pheromone message. When the robot detects the wall, it changes modes to avoid "wall" pheromone while moving towards the wall in order to park. This results in a fairly even distribution of robots along the wall.

3.5.3 Cooperative Sensing

Cooperative sensing is an even higher level compound behavior built upon gradient following. Positive identification of some objects requires agreement between two or more robots, either to provide redundancy for fault tolerance, or to raise confidence through cross-correlation. In many cases, tiny robots with limited power and sensing ability must specialize. In the demo shown in figure 8, we portray a scenario in which a few robots have acoustic sensors, a few others have motion sensors, and the remaining robots have no specialized sensors (the actual robots can only detect beacons, so the specialized sensors were simulated). Reliable detection of intruders in this example requires confirmation from both a motion sensor and an acoustic sensor in the same location. When an acoustic sensing robot detects something, it initiates a pheromone gradient. Motion sensing robots are attracted to follow this gradient. But, since we do not want every motion sensing robot to head toward the same location, a winner-take-all strategy is used to allow only the closest robot to approach the objective. The motion sensing robots

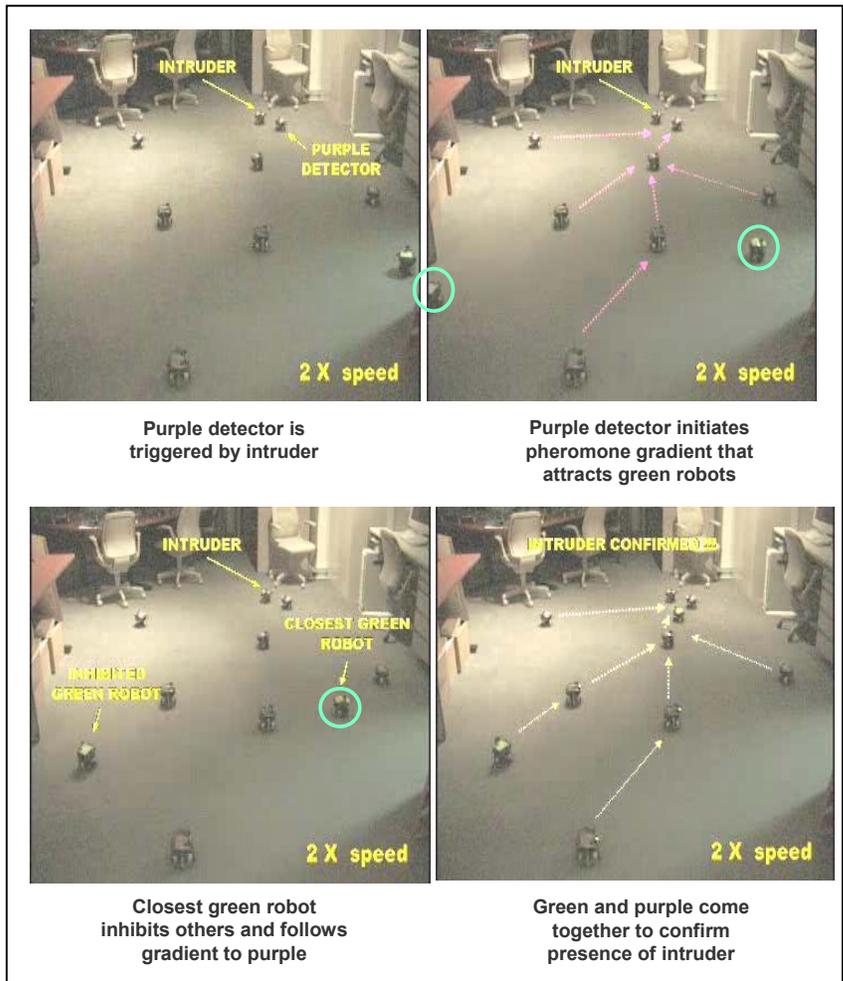


Figure 8: Four stages in a cooperative sensing task.

transmit their own pheromone signal, which includes a dominance score, derived from the gradient hop-count, indicating how close they are to the objective. Motion-sensing robots with lower scores are thus inhibited so that only the closest one follows the gradient. This provides an inherent stability in the selection of responding robots because the dominant robot only becomes more dominant as it moves up the gradient. If for any reason, the dominant robot should fail or be destroyed, it will no longer transmit its inhibitory signal, and another similarly equipped robot will immediately take over. When the motion-sensing robot finally reaches the acoustic sensing robot, it too will detect the intruder and initiate its own gradient. With both acoustic and motion detections co-located, neighboring robots detecting both gradients trigger an intruder alert that propagates throughout the robot swarm.

4. Conclusion

We have demonstrated collections of robots performing complex tasks such as leading the way through a building to a hidden intruder or coordinating the movement of distributed sensors to confirm a detection. This is possible because the robot collective becomes a computing grid embedded within the environment, while also acting as a physical embodiment of the user interface. Over the past decades, the literature on path planning and terrain analysis has dealt primarily with algorithms operating on an internal map containing terrain features. Our approach *externalizes* the map, spreading it across a collection of simple processors, each of which determines the terrain features in its locality. The terrain processing algorithms of interest are then spread over the population of simple processors, allowing global computations such as path planning to be performed by the population. We have extended these computations to perform other tasks such as resource allocation and coordinated movement.

We have shown how complex compound behaviors can be composed from combinations of simpler behaviors. Also we have shown how even the simpler behaviors can be composed from combinations of movement and pheromone logic primitives. For example, gradient following is composed from gas expansion, gradient propagation, and pheromone-specific attraction primitives. We build upon this as an element in the cooperative sensing behavior, which uses gradient following, inhibition, and logic for detecting the simultaneous presence of two different pheromone types. In general it is difficult to say whether or not a complete set of primitives has been produced; however, our methodology provides us with an effective building block approach to constructing compound group behaviors.

The approach to pheromone robotics described in this paper offers robustness because it needs no explicit maps or models of the environment, and requires no explicit knowledge of robot location. The user interface to this distributed robot collective is itself distributed. Instead of communicating with each robot individually, the entire collective works cooperatively to provide a unified display embedded in the environment. The combination of this world-embedded interface with our world-embedded computation means that the results of complex distributed computations can be mapped directly onto the world with no intermediate representations required.

5. Acknowledgements

This work is supported by the Defense Advanced Research Projects Agency under contract N66001-99-C-8514. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Defense Advanced Research Projects Agency.

6. References

- [1] R.C. Arkin, *Behavior-Based Robotics*. MIT Press, Cambridge, MA. 1998.
- [2] R.C. Arkin and G.A. Bekey, (editors), *Robot Colonies*. Kluwer Academic Publishers, 1997.
- [3] R. Azuma, B. Hoff, H. Neely III, R. Sarfaty, A Motion-Stabilized Outdoor Augmented Reality System, *Proc. IEEE VR '99*, Houston, TX, 1999, pp. 252-259.
- [4] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. New York, Oxford University Press 1999.
- [5] J. Deneubourg and S. Goss, Collective Patterns and Decision-Making, *Ethology, Ecology, and Evolution*, 1984, 1:295-311.
- [6] E.W. Dijkstra A Note on Two Problems in Connection with Graph Theory, *Numerische Mathematik*, 1959, 1:269-271.
- [7] D.W. Gage, 1992. Command and Control for Many-Robot Systems, In *Unmanned Systems Magazine*, 10(4):28-34.
- [8] D.W. Gage, 1993. How to Communicate with Zillions of Robots, In *Proc. SPIE Mobile Robots VIII*, Boston, MA, 2058:250-257.
- [9] S. Goss, R. Beckers, J. Deneubourg, S. Aron, and J. Pasteels, 1990. How Trail Laying and Trail Following Can Solve Foraging Problems, In *Behavioral Mechanisms of Food Selection*, ed. R. Hughes, Springer-Verlag, Heidelberg, Germany, pp. 661-678.
- [10] O. Holland and C. Melhuish, Stigmergy, self-organization, and sorting in collective robotics. *Artificial Life*, 5(2), 2000.
- [11] M.A. Lewis, and G.A. Bekey, The Behavioral Self-Organization of Nanorobots Using Local Rules," In *Proc. 1992 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Raleigh, NC 1992.
- [12] J.S.B. Mitchell, D.W. Payton, and D.M. Keirse, Planning and Reasoning for Autonomous Vehicle Control, *Int J. Intelligent Systems*, Vol. 2, 1987.
- [13] D.W. Payton, Internalized Plans: A Representation for Action Resources, in *Designing Autonomous Agents*, ed. Pattie Maes, MIT Press, Cambridge, Mass, 1990, pp. 89-103.
- [14] D. Payton, M. Daily, R. Estkowski, M. Howard, C. Lee, "Pheromone Robotics", in *Autonomous Robots*, Kluwer Academic Publishers, Boston, MA, 11(3) 2001, pp. 319-324.
- [15] C. Ünsal, and J. Bay, Spatial Self-Organization in Large Populations of Mobile Robots," *IEEE Int. Symp. on Intelligent Control*, 1994, pp. 249-254.
- [16] Teambots Web site: <http://www.teambots.org/>
- [17] B.B. Werger, and M.J. Mataric, Robotic food chains: Externalization of state and program for minimal-agent foraging, In *Proc. 4th Int. Conf. Simulation of Adaptive Behavior: From Animals to Animats 4*, MIT Press, 1996, pp. 625-634.