J Intell Robot Syst (2006) 45: 307–321
DOI 10.1007/s10846-006-9038-4

Using Virtual Pheromones and Cameras for Dispersing a Team of Multiple Miniature Robots

Janice L. Pearce · Bob Powers · Chistopher Hess · Paul E. Rybski · Sascha A. Stoeter · Nikolaos Papanikolopoulos

Received: 11 April 2005 / Accepted: 11 April 2006

Abstract To safely and efficiently guide personnel of search and rescue operations in disaster areas, swift gathering of relevant information such as the locations of victims, must occur. Using the concept of 'repellent virtual pheromones' inspired by insect colony coordination behaviors, miniature robots can be quickly dispersed to survey a disaster site. Assisted by visual servoing, dispersion of the miniature robots can quickly cover an area. An external observer such as another robot or an overhead camera is brought into the control loop to provide each miniature robot estimations of the positions of all of the other near-by robots in the robotic team. These miniature robots can then move away from the other near-by robots on the team, resulting in the robot collective becoming swiftly distributed through the local area. The technique has been simulated with differing pheromone persistence levels and implemented using the miniature Scout robots, developed by the Center for Distributed Robotics at the University of Minnesota, which are well-suited to surveillance and reconnaissance missions.

Key words Dispersion · distributed robotics · mobile robots · miniature robotics

1. Introduction and Problem Description

Remote surveillance and reconnaissance applications frequently make use of multiple remote sensing devices that report back to a human or robot coordinating agent. When this technique is employed, the coordination agent needs to assure that

J. L. Pearce · B. Powers · C. Hees
The Department of Mathematics and Computer Science, Berea College, Berea, USA

P. E. Rybski · S. A. Stoeter · N. Papanikolopoulos (✉)
The Department of Computer Science, University of Minnesota, 200 Union St. SE, 4192 EE CSC Building, Minneapolis, MN 55455, USA
e-mail: npapas@cs.umn.edu

Figure 1 Two Rangers with seven Scouts.

adequate sensor coverage is attained. If the application is time-critical such as is often the case at a disaster site, then this sensor coverage must be achieved quickly.

One approach to operating in this scenario is to employ a mobile robotic or human agent that is capable of long distance travel to coordinate, deploy, and communicate with multiple smaller sensing agents. The coordination agent can then query the smaller sensing agents for information and move them remotely to increase the area surveyed. The Center for Distributed Robotics at the University of Minnesota has developed such a team of robots. Larger 'Ranger' robots which are capable of navigating long distances over rough terrain without needing to recharge their batteries are used to traverse the environment and to deploy the miniature 'Scout' robots (shown in Figure 1). Equipped with a magazine and a spring-based delivery mechanism, a Ranger can deliver up to 10 Scouts into a target area. The Ranger's more powerful onboard computer can then be used to coordinate Scouts and relay status information [14].

A need for a fast dispersal technique exists regardless of whether the coordinating agent is a human or a robot. When the coordinating agent is a human equipped with a wearable computer interface, commands can be sent to the robots from a hand-held controller and sensor information can be returned to the human through a head-mounted display. However, a human operator cannot be expected to simultaneously control each robot individually during deployment, since the operator's attention must be completely dedicated to teleoperation of a single robot. Thus, the work described in this article is designed to create a method for efficiently deploying and dispersing robots such as the Scout robots regardless of the type of coordinating agent. The robots as soon as they are out of the observer's field-of-view might employ visual servoing techniques in order to navigate towards their desirable goal.

2. Related Work

Team behaviors have been studied in a variety of disciplines from the biological studies of herds and swarms to the sociological studies of societies of humans. Physicists and chemists have studied the behaviors of collections of a variety of interacting bodies from gravitational planetary forces to the movements of various particles. Many of these studies of interactions in the natural world have become models for the behaviors of teams of robots, particularly as the robotic teams engage in tasks such as dispersion and area coverage.

In 1992, Gage categorized the concept of ‘area coverage’ by a robotic team into three basic types of coverage: ‘Blanket coverage,’ in which the main objective is to maximize the total detection area; ‘barrier coverage,’ where the objective is to minimize the possibility of undetected penetration of a defined barrier; and ‘sweep coverage,’ where the objective is to cover an area with a sweeping or moving barrier [3]. Using this taxonomy, the objective of the work described in this paper is to quickly deploy robots and achieve either a blanket coverage or a circular sweep coverage of an area.

In 1992, Gage also designed some robot coordination simulations such as ‘condensation’ based loosely upon biosystem analogies such as pheromones [3]. In the early work by Arkin and Ali, the dispersion of a robotic team was carried out by a random-wandering behavior coupled with moderate robot repulsion as well as more significant obstacle repulsion [1]. In 1995, directly inspired by animal navigation routines, M. Matarić and her research group designed a dispersion algorithm that moves an agent away from the centroid of the local density distribution of the other agents that are visible to that agent’s sensors [7].

In 1999, Spears and Gordon provided distributed control of large collections of agents by having agents react to artificial forces motivated by natural laws of physics, observing that in the real physical world surprisingly complex behaviors arise from simple interactions between entities. However, their applications were self-assembly and self-repair rather than dispersion for the purpose of surveillance [15]. In another virtual physics approach, Howard et al. used a ‘potential-field-based approach’ to the deployment of a mobile sensor network by treating their robots as virtual particles subjected to virtual forces [5]. These forces cause each given robot to be repelled from the other robots as well as from other obstacles in the environment with a potential that is proportional to the sum of the reciprocals of the distances from the first given robot. Though this portion of the algorithm is somewhat similar to the work presented in this paper, Howard et al. continue to run their algorithm until the network as a whole reaches a state of static equilibrium while in this paper after the initial dispersion, other robot behaviors such as locating a specific goal are allowed to operate [5].

The work that perhaps shares the most motivational similarities with the techniques described in this paper is the research of Payton et al. which employs techniques for coordinating the actions of large numbers of small-scale robots used in surveillance, reconnaissance, hazard detection, and path finding [8]. As in our project, they exploit the biologically inspired notion of a ‘virtual pheromone,’ but they implement their virtual pheromones using transceivers mounted atop each robot rather than with global information from an overhead camera [8, 9].

Batalin and Sukhatme also address the problem of multi-robot area coverage from the premise that local dispersion of robots will ultimately achieve good global area coverage [2]. As in this paper, their algorithms result in their robots being ‘mutually repelled’ from one another, however like Payton et al., they depend upon their robots to be able to sense or recognize one another rather than on global information such as from an overhead camera.

Stoeter et al. effectively use an overhead camera to track and direct a miniature Scout robot marked with color-markers as it travels, orients on a target, and climbs stairs. Though the extension to the problem of multiple robots is discussed in the article, experiments are carried out with a single robot [17].

3. Modeling with Repellent Pheromones

Pheromones are chemicals used in nature as a form of indirect communication that can produce organized group activities. For example, ants leave a trail of pheromone to mark the path that they traverse between their nest and a food source. As more ants traverse this path, the pheromone trail is reinforced. The main purpose of this paper is to model ‘repellent pheromones’ in order to bring about the dispersion of a robotic team. Many approaches to dispersion require prior knowledge of the deployment area, but this approach has the advantage that it requires no map of the area and the robots need no self-knowledge of their location within the area. Other approaches to dispersion use a global geometric model, while our approach is based only upon decisions that are made locally. This has the distinct advantage of flexibility; suppose one robot loses its ability to travel, the local approach will automatically adjust to the given situation while the global approach would require a complete reworking.

Virtual pheromones should degrade as the distance from the virtual pheromone emitter increases. Since a robot may be subjected to multiple virtual pheromone emitters from different directions, we model the cumulative repellent force of the virtual pheromones as a vector sum with the vector length given by a decreasing function. How quickly the virtual pheromones degrade as distance increases determines which decreasing function is employed. For our vector length, we chose to use a power of the reciprocal of the distance the virtual pheromone travels. This choice is motivated by natural phenomena such as electric field strength, which is inversely proportional to the distance from a charged object. Simulations have been run by varying persistence of the pheromone over distance by varying the power on the reciprocal of the distance and runs have been carried out with unit power.

In particular, if $\boldsymbol{x}$ is the position of robot R, then $p_i = \frac{1}{\|\boldsymbol{x}-\boldsymbol{x}_i\|^l}$ will model the level of virtual pheromone emitted by robot R_i that is detected by robot R. Here the positive number l is called the localization factor and is used to vary the persistence level of the virtual pheromone. Clearly, increasing l will cause the pheromone to have a more localized effect on nearby robots because the strength of the pheromone will fall off more rapidly over distance. The direction of this virtual pheromone is $\boldsymbol{d}_i = \frac{\boldsymbol{x}-\boldsymbol{x}_i}{\|\boldsymbol{x}-\boldsymbol{x}_i\|}$. Thus, the total repellent direction of all of the detected virtual pheromones is $\sum_i p_i \boldsymbol{d}_i = \sum_i \frac{\boldsymbol{x}-\boldsymbol{x}_i}{\|\boldsymbol{x}-\boldsymbol{x}_i\|^{(l+1)}}$.

With n robots in randomly distributed starting positions, iterating this algorithm infinitely many times on an infinite plane, will, in nearly all cases, direct the robots to positions that approximate the vertices of a regular n-gon, thus asymptotically approaching the perfect circular sweep coverage of the area. In situations with more pathological starting positions, such as with three or more robots whose centroids are exactly in a line, the noise inherent in the physical system should cause one or more of the robots to move off of the line and then asymptotically approach perfect circular sweep coverage. In simulations as the localization parameter l is increased, robots achieve a given approximation to perfect circular sweep coverage more quickly. This can be seen by considering the standard deviations of the robots' nearest neighbor distances.

In our simulation work, we employ velocity vectors for the trajectories of our simulated robots. Since velocity is the change in position over the change in time, we consider velocity as a derivative and approximate the derivative via Euler's method. In particular, the velocity equations yield a set of differential equations that is solved to obtain the new position vector for each robot.

As expected, increasing the localization parameter l that represents the virtual pheromone's persistence over distance causes the dispersion of the robots to be decreasingly effected by robots that are farther away than those nearer.

4. Results and Analysis

4.1. Color Tracking of Scout Robots

A robot such as a Scout that is small enough to avoid detection and can access hard-to-reach areas is extremely useful for surveillance and reconnaissance applications, but the small size brings certain challenges to the task of using color markers for tracking. The Scout robot (shown in Figure 2) is cylindrical, measuring only 11 cm long and 4 cm wide. The electronics of Scouts include microcontrollers, transmitters, magnetometers, tiltometers, and shaft encoders. The Scout has differentially driven wheels and a leaf-spring tail jumping mechanism. Scouts also carry a sensor payload, usually a miniature video camera, used to broadcast environmental information over an analog RF transmitter. For color tracking, two color markers are needed to be as widely separated as possible on each Scout robot in order to obtain accurate location and orientation information, but the markers must not interfere with the workings of the Scout. For this reason, color bands of approximately 1 cm in length encircle only the two opposite ends of the deployed Scouts.

By quickly providing information on the relative locations and orientations of each of the Scout robots at the point of deployment, an overhead camera and standard vision techniques are used to assist in the dispersion of the robots. Each of the color markers is tracked as a color 'blob,' and the centroids of these blobs are averaged

Figure 2 Two Scout robots aside a ruler for scale.

to determine the location of the centroid of the Scout robot itself. The Scout's orientation is a vector in the direction of the forward direction of locomotion for the Scout. This is computed by rotating 90° from the vector along the Scout from the centroid of the left-marker to the centroid of the right-marker.

The overhead camera used for the experiments was a Panasonic GP-KR222 model with 480 lines horizontal resolution and a minimum scene illumination of 3 lux at $f/1.4$ (or 2 lux at $f/1.2$). The issue with color tracking is that it typically needs fairly good lighting conditions to work properly. If you are using nicely saturated colors, as the light becomes dimmer, these colors become darker and fall out of the range of what might be distinguished easily by a lookup table. Initially the color-tracking for this experiment was conducted using the same techniques and software as Stoeter et al. [18], however though the technique worked well for a single Scout robot, it proved not to be easily scalable. In [18], the ends of the Scout were also marked with different colors to aid pose estimation. Smoothing was accomplished with a 5×5 median filter and then converted from the RGB to the HLS color model. Thresholding of the image took place with the known ranges of hue, lightness, and saturation from both markers. A circle was fit to the largest blob in the thresholded image for each region of interest in order to be more sure of finding the correct blob. Using the Stoeter technique, hand-selected color ranges in the HLS color space were used to identify each color marker. This worked very well to segment widely spaced colors in the HLS color space as red and green. However, as the number of color-markers increased, it became increasingly difficult to create disjoint color ranges.

The difficulties with glare and changing lighting conditions, caused the sizes and positions of the detected blobs to vary widely. Using these statistics in the calculations of Scout positions served to amplify the difficulties. In order to test the reliability of the range-based color selection system, color-tracking was performed with immobile Scouts. As can be seen in Table I, the error in orientation was untenable; with an error of 142°, even an optimal algorithm for dispersion would have seemed to give only random results.

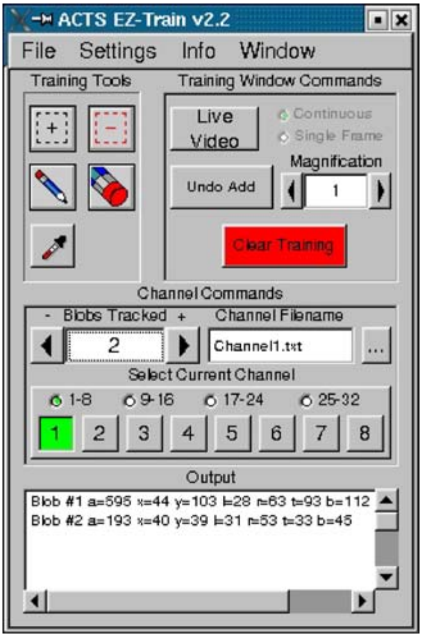
Finding ways to address problems of noise is crucially important since any perceptual system which operates in the real world must be able to recognize and correct for corrupted sensor data if it expects to operate correctly [4]. In related work [13], Rybski et al. used a simple frame averaging algorithm to reduce the effects of noise, but this approach could not be used on data that was as poor as in Table I.

In order to achieve accurate tracking, we adopted the use of the ActivMedia Color Tracking System (ACTS) [12] for performing color thresholding and computing blob statistics (interface shown in Figure 3). ACTS allows users to select regions of color in an image not simply as ranges, but almost as finely as selected sets of points in RGB space. In fact, ACTS stores the projections of the RGB voxels onto the three planes Red-Green, Blue-Green, Red-Blue, giving far more flexibility than a simple sub-cube

Table I Deviations in Scout-tracking using ranges in HLS color space

Using HLS ranges	Scout 1	Scout 2	Scout 3
Standard deviation in x position (pixels)	17.7	5.9	26.9
Standard deviation in y position (pixels)	7.0	3.0	12.6
Standard deviation in orientation (degrees)	27.5°	142°	71.7°

Figure 3 ACTS interface.



in RGB space. These color regions are then tracked as blobs, and statistics are computed for each blob. ACTS users can train color channels by selecting individual pixels of color from the image by clicking on a window to select the color markers directly. The ACTS-trained color channel files are stored as look-up table files. ACTS is able to track 32 individual color channels simultaneously at greater than 30 fps on a Linux Pentium 160 MHz, so its speed and number of channels are more than sufficient for our application since the limiting factor in our dispersion speed proved to be the RF command speed.

Using ACTS with a window size of only 320 pixels $\times$ 240 pixels and the above-described technique for color-tracking gave us average standard deviations of Scout positions of less than two pixels and of orientations of less than 3° for a set of up to eight selected colors (see Table II). We elected to track only blobs larger than four pixels wide, since smaller blobs are likely to be simply noise due to such issues as the combined effects of fluorescent and natural lighting conditions changes.

4.2. Radio-Controlling the Scout Robots

After implementing color-tracking for detecting current robot locations and orientations and modeling with repellent virtual pheromones, the next step is to direct

Table II Color markers used for color-tracking of up to four Scouts

Color markers	Scout 1	Scout 2	Scout 3	Scout 4
Right marker	Neon orange	Sky blue	Dark green	Teal blue
Left marker	Neon green	Yellow	Deep purple	Neon pink

the movements of the robots. Due to the small size and power constraints of the current version of the Scout robots, very limited on-board computational power is available, since they require their two CPUs for network communications and actuator control. Thus, intelligence for control decisions must be provided externally. This implementation of dispersion of the Scout robots involves external visual servoing [6] and requires the auxiliary hardware of a computer equipped with a framegrabber to run the image-processing algorithms. This computer could be either a Ranger or another machine within reception range of the Scout's analog video transmission.

To overcome the Scouts' size-imposed limitations and to connect multiple computers for complex missions, a distributed software architecture has been developed that supports the transparent integration of remote resources [16]. A functional view of missions is taken, so all hardware resources, including the robots, are partitioned into finely grained resources that can be requested by functional components.

The architecture consists of the four subsystems shown in Figure 5:

- The *User Interface* subsystem provides run-time control and feedback.
- *Mission Control* hosts prioritized components that, together, make up a solution for a mission. Components are only partially-ordered, so many can execute in parallel.
- The *Resource Management* contains resources, such as cameras and transmission frequencies, that can be requested by components.
- The *Backbone* provides basic services that connect the other three subsystems transparently over all processors available for a mission [16].

This distributed software control architecture dynamically coordinates hardware resources and shares them between the various clients, allowing for simultaneous control of multiple robots. Each Scout has a unique network ID which allows commands to be routed to specific robots while being ignored by others allowing multiple robots to be controlled from a single RF modem. Motion commands are

Figure 4 Connecting a wearable processor to the user interface and Scout radios.

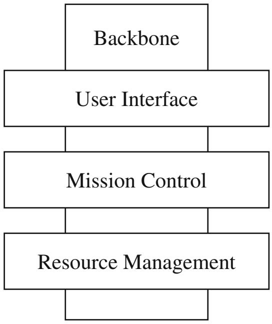


Figure 5 The core of the distributed software architecture is comprised of four subsystems.

then transmitted from a remote source and are received and executed by the Scout robot.

As shown in Figure 4, the processor can be portable and can exist between the user interface and the wearable controller. The computer receives inputs from the operator and sends the appropriate commands to the radio. Received video is captured by a framegrabber and is presented to the user via a head-mounted or an arm-mounted display. This captured video can also be used by autonomous Scout control programs running on the processor (Figure 5).

However, due to noise in the system such as radio interference, Scouts do not always receive the commands that are sent to them. Even when they do receive the intended commands, they may not receive the commands for the duration that the command was intended. These issues cause control difficulty, particularly in finely adjusting the pose of the Scout, since turning by varying degrees is accomplished by directing the Scout to rotate one or both wheels for a certain duration of time.

Orienting the body of a Scout so that it faces a target head on is an important task apart from the dispersion problem. In a variety of situations, the Scout needs to be aligned correctly. Such scenarios could include docking with a larger robot such as the Ranger for pickup, or for using landmarks for tasks such as localization. Previous experiments in tracking Scouts have utilized pattern matching [11] and active contour models [10], but neither approach proved completely adequate for the given task. The inaccuracies can be seen in Tables III, IV, V, and VI.

Table III Trials for turning Scouts by a given angle

Requested turn angle	Average turn error	Communication failures
270°	3.1°	2
180°	4.6°	1
90°	10.1°	0
45°	5.9°	2
30°	3.3°	1
15°	3.1°	1

Two Scouts were used simultaneously for each trial. The average turn error was averaged over five runs. Communication failures indicates the number of times a Scout failed to receive commands and to move during an experiment.

Table IV Trials for turning Scouts for a given large angle

Trial: 270°		Trial: 180°	
Scout 1	Scout 2	Scout 1	Scout 2
270°	No response	175°	175°
270°	265°	182°	175°
278°	272°	No response	170°
268°	266°	180°	178°
No response	266°	178°	170°
Average deviation: 3.1°		Average deviation: 4.6°	

Scouts are able to receive radio signals on two different radio frequencies. However, in this work, all of the Scout commands are sent on a single one of these two frequencies, because the single serial port of the processing computer is used to direct the commands to the radio. The speed of the radio commands proved to be the limiting factor in the speed of dispersion, because commands sent too quickly in succession interfered with each other and caused the Scout to fail to be able to discern the commands sent. Thus, a pause of between 150,000 and 300,000 μs was added to slow the command rate. In addition because commands to different Scouts are sent on the same frequency, only a single Scout is actually able to receive a command at a given time instant (Table VII).

4.3. Dispersing the Scout Robots

Dispersion runs have been completed using the virtual pheromone model described above with one, two, three, and four Scouts (where one Scout is included only for testing the robustness of the algorithm). Given the unreliability of the radio signal being correctly received, the results were surprisingly good. In Table VIII, results are tabulated as deviations from optimal beginning with the three Scouts poses of (0°, 0°), (0°, 90°), and (0°, 180°) (Figure 6). Error runs with more Scouts show similar errors.

Figure 7 shows the actual dispersion paths taken by the Scouts during various runs. One notes that while some of the Scouts choose their direction quickly and drive away, it is possible to see the adjustments of other Scouts as they correct their turns. In the run with four Scouts, it is also possible to note an error in the color tracking. The Scout in the upper right-hand corner of Figure 7e did actually head the wrong direction and then retrace its path exactly. This error in the blob detection

Table V Trials for turning Scouts by a given medium angle

Trial: 90°		Trial: 45°	
Scout 1	Scout 2	Scout 1	Scout 2
100°	90°	50°	46°
93°	108°	52°	55°
98°	108°	No response	45°
92°	102°	55°	54°
101°	95°	No response	50°
Average deviation: 10.1°		Average deviation: 5.9°	

Table VI Trials for turning Scouts by a given small angle

Trial: 30°		Trial: 15°	
Scout 1	Scout 2	Scout 1	Scout 2
28°	31°	20°	15°
28°	29°	No response	18°
24°	25°	17°	18°
34°	No response	22°	10°
30°	21°	17°	16°
Average deviation: 3.3°		Average deviation: 3.1°	

occurred for a single frame and so did not cause the Scout to traverse an incorrect dispersion path in the end. Though such errors are infrequent, they are obvious when viewing the data visually. Even given these issues, the trajectories of the Scouts create a reasonable dispersion in each case.

As for a measure of team performance, one immediate limitation that comes to mind is the Field-Of-View (FOV) of the camera. If the camera does not have a very wide angle fish-eye lens (which will introduce other problems with color tracking and chromatic aberrations), then the area under which the robots can travel will be quite small. We also tried to implement some metric that involved the amount of space in the FOV that was covered, or something to the effect of how far the Scouts reached to the perimeter of the FOV, given the fact that they started in the center. That is, how many reached the perimeter in a certain amount of time? However, we found out that the dispersion errors were the best indicators given the FOV of the camera employed.

This project explored the dispersion of a robotic team such as might be used for reconnaissance and surveillance applications designed to operate in a semi-autonomous fashion. A human operator is able to remotely direct the robot to disperse in unknown areas and then allow the robot to do some of the tasks autonomously. However, the next generation of Scout robot currently under development will be much larger and more powerful with much more processing power on-board, so more decision-making will be able to occur on-board these future Scouts without using the communication channels. Since the communication has proved to be the factor that solely determines the speed of the dispersion in this work, reducing the need for radio communication should dramatically speed up the dispersion of the robots.

For our actual runs, repellent pheromones are modeled with a repellent force given by the reciprocal of the distance the pheromone has traveled. We have also completed simulations on robot dispersion behavior using pheromones that degrade

Table VII Dispersion errors with two Scouts from a given pose

Starting poses	Average turn error	Communication failures
(0°, 0°)	4.6°	1
(0°, 90°)	5.7°	0
(0°, 180°)	6.7°	0

Five runs were done for each starting pose. Communication failures indicates the number of times a Scout failed to receive commands and move during an experiment.

Table VIII Dispersion errors with two Scouts from a given pose

Start pose: (0°, 0°)		Start pose: (0°, 90°)		Start pose: (0°, 180°)	
Scout 1	Scout 2	Scout 1	Scout 2	Scout 1	Scout 2
0°	No response	8°	+1°	−6°	3°
−3°	−8°	−3°	+5°	+5°	−5°
+4°	−10°	−2°	+15°	−3°	2°
+2°	−7°	−4°	−10°	+8°	−10°
0°	+9°	−2°	+7°	−20°	+5°
Avg. deviation: 4.6°		Avg. deviation: 5.7°		Avg. deviation: 6.7°	

more and less quickly over distance, so in our future work, we will implement these differing pheromone persistence levels with our miniature robot team. Following the simulation runs which used localization parameters of $l = 1$ and $l = 2$ with four robots started from various initial positions, and allowing for differing dispersal times, and differing robot speeds, the standard deviations of the distances between each robot's two nearest neighbors were computed. In simulation, the higher localization parameter generally lead to better approximation to perfect circular sweep coverage. It is not clear how noise in the system will affect these simulation results, so we plan to implement future test runs on the robot team with different localization parameters.

In this work, the Scout's location and orientation are calculated from vision-analysis of the position of the colored markers, then the pheromones are modeled virtually. While the color-tracking analysis has proved quite accurate using the ACTS software, improved results might be achieved by applying a Kalman filter. The Scout's color markers could be supplemented and/or replaced by colored wheels, possibly yielding an increase in tracking accuracy because the markers would be larger and farther apart. It would also allow observers on the ground to better track Scouts from the side. However, in place of the color-tracking implementation, the implementation of virtual pheromones by using a short-range transceiver should be considered even though it would require additional on-board power as it would offer the following additional benefits:

- Transceiver-implemented virtual pheromones would operate anywhere the Scouts were operating, so no overhead camera or Ranger-mounted camera would need to be present.

Figure 6 Scout poses of **a** (0°, 0°), **b** (0°, 90°), and **c** (0°, 180°).

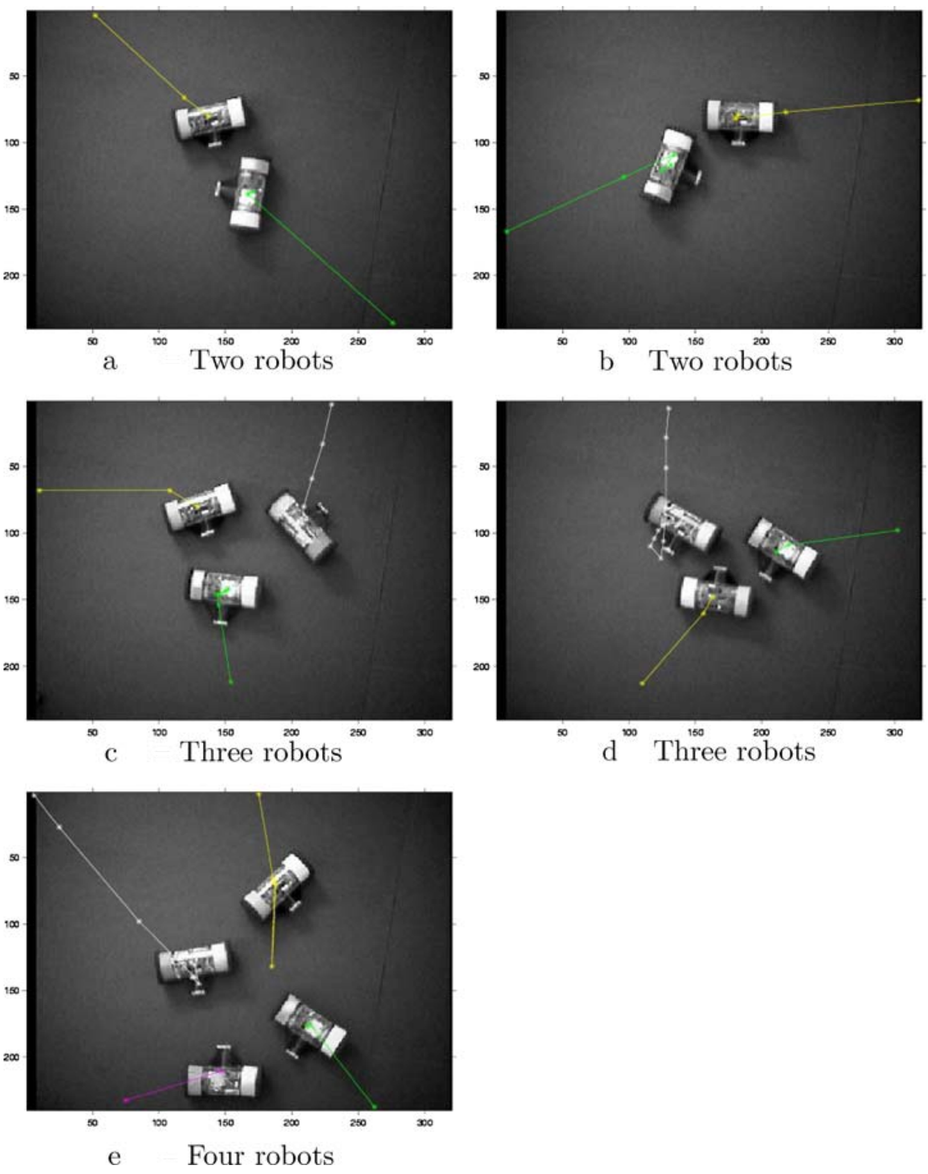


Figure 7 Example dispersion runs with Scout robots (axes in pixels).

- When a Scout power supply becomes exhausted, the color-markers still appear on the Scout, but a transceiver would stop broadcasting. Thus, a Scout that is ‘down’ would disappear and other Scouts would move in to cover the area.
- Obstacles that block a Scout’s view would likely also block the transceiver signal, so coverage of areas with short obstacles would likely be improved.

- In addition to dispersion (and grouping), virtual pheromones implemented with a transceiver could be employed in additional applications such as finding a shortest path though a maze-like site.
- With more processing power available on-board and transceiver-implemented virtual pheromones, the decision-making of the next generation of Scout robots can be much more distributed, including dispersal without use of the communications channel.

Future work will expand on the Scout's autonomous capabilities, which will include more advanced sensor interpretation and spatial reasoning techniques. The software control architecture is also being expanded to allow more types of hardware resources, such as larger robots, to be controlled.

5. Conclusions

The approach of using repellent virtual pheromones as described in this paper offers a fairly robust approach to the dispersion of a robotic team. It needs no prior map of the area and requires no localization, yet it leads to a reasonable implementation of a dispersion for broadcast coverage, even when noise is present in the system. Though improvements can certainly be made in future implementations, this technique can be implemented as is for the dispersion of robots.

Aknowledgments This material is based upon work supported by the Defense Advanced Research Projects Agency, Microsystems Technology Office (Distributed Robotics), ARPA Order No. G155, Program Code No. 8H20, issued by DARPA/CMD under Contract #MDA972-98-C-0008, the Microsoft Corporation, INEEL, and the NSF under grants #CNS-0224363, #CNS-0324864 and #CNS-0480326. We would like to thank the anonymous reviewers for their thoughtful comments.

References

1. Arkin, R., Ali, K.: Integration of reactive and telerobotic control in multi-agent robotic systems. In: Proc. Third International Conference on Simulation of Adaptive Behavior, (SAB94) [From Animals to Animats], pp. 473–478. Brighton, England (1994)
2. Batalin, M., Sukhatme, G.: Spreading out: A local approach to multi-robot coverage. In: The 6th International Symposium on Distributed Autonomous Robotic Systems, pp. 373–382. Fukuoka, Japan (2002)
3. Gage, D.: Command control for many-robot systems. In: AUVS-92, the Nineteenth Annual AUVS Technical Symposium. Reprinted in Unmanned Systems Magazine, Fall 1992, Volume 10, Number 4, pp. 28–34. Huntsville, Alabama (1992)
4. Hoffman, R., Krotkov, E.: Terrain mapping for outdoor robots: Robust perception for walking in the grass. In: Proc. of the IEEE Int'l Conf. on Robotics and Automation, 529–533, 1993
5. Howard, A., Matarić, M., Sukhatme, G.: Network deployment using potential fields: A distributed, scalable solution to the area coverage problem. In: The 6th International Symposium on Distributed Autonomous Robotic Systems, pp. 299–308. Fukuoka, Japan (2002)
6. Hutchinson, S., Hager, G.D., Corke, P.I.: A tutorial on visual servo control. IEEE Trans. Robot. Autom. **12**(5), (1996)
7. Matarić, M.J.: Designing and understanding adaptive group behavior. Adapt. Behav. **4**(1), 51–80 (1995)
8. Payton, D., Daily, M., Estkowski, R., Howard, M., Lee, C.: Pheromone robotics. Auton. Robot. **11**(3), 319–324 (2001)
9. Payton, D., Estkowski, R., Howard, M.: Progress in pheromone robotics. In: 7th International Conference on Intelligent Autonomous Systems, 2002

10. Perrin, D.P., Kadioglu, E., Stoeter, S.A., Papanikolopoulos, N.: Localization of miniature mobile robots using constant curvature dynamic contours. Proc. of the IEEE Int'l Conference on Robotics and Automation, 2002
11. Rößler, P., Stoeter, S.A., Rybski, P.E., Gini, M., Papanikolopoulos, N.: Visual servoing of a miniature robot toward a marked target. In: Proc. of IEEE International Conference on Digital Signal Processing, July 2002
12. Rybski, P.E.: ACTS: Activmedia Color Tracking System Manual. ActivMedia Robotics, LCC, 44 Concord Street, Peterborough, New Hampshire 03458, USA (2002)
13. Rybski, P.E., Papanikolopoulos, N., Stoeter, S.A., Krantz, D.G., Yesin, K.B., Gini, M., Voyles, R., Hougen, D.F., Nelson, B., Erickson, M.D.: Enlisting rangers and scouts for reconnaissance and surveillance. IEEE Robot. Autom. Mag. **7**(4), 14–24 (2000)
14. Rybski, P.E., Stoeter, S.A., Erickson, M.D., Gini, M., Hougen, D.F., Papanikolopoulos, N.: A team of robotic agents for surveillance. In: Proc. of the Int'l Conf. on Autonomous Agents, pp. 9–16. Barcelona, Spain (2000)
15. Spears, W., Gordon, D.: Using artifficial physics to control agents. In: IEEE International Conference on Information, Intelligence, and Systems, 1999
16. Stoeter, S.A., Rybski, P.E., Erickson, M.D., Gini, M., Hougen, D., Krantz, D.G., Papanikolopoulos, N., Wyman, M.: A robot team for exploration and surveillance: Design and architecture. In: Proc. of the Int'l Conf. on Intelligent Autonomous Systems, pp. 767–774. Venice, Italy (2000)
17. Stoeter, S.A., Rybski, P.E., Gini, M., Papanikolopoulos, N.: Autonomous stair-hopping with scout robots. In: Proc. of the IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, 2002
18. Stoeter, S.A., Rybski, P.E., Stubbs, K.N., McMillen, C.P., Gini, M., Hougen, D.F., Papanikolopoulos, N.: A robot team for surveillance tasks: Design and architecture. Robot. Auton. Syst. **40**(2–3), 173–183 (2002)