Dispersion of a swarm of robots based on realistic wireless intensity signals

Emre Ugur, Ali E. Turgut, Erol Şahin
Kovan Research Lab., Dept. of Computer Eng.
Middle East Technical University, Ankara, Turkey
{emre,ali,erol}@ceng.metu.edu.tr

Abstract— **The problem of dispersion in multi-robot systems could be loosely defined as maximizing the sensor coverage area while preserving the connectivity within the swarm. Dispersion of robotic swarms appears to be applicable and useful in missions such as planetary exploration, hurricane surveillance, or nuclear decontamination, where the robots with maximal coverage collect samples from the unknown surface, detect the victims, or collect nuclear waste, respectively. In this paper, a simple dispersion algorithm based on wireless signal intensities is proposed and tested in a physics based simulator of a robotic platform which is particularly designed to serve as a test-bed for swarm-robotic studies. The signal intensities are realistically modeled using sampling technique, taking both the distance and relative orientations of the wireless sensors into account. The only parameter of the algorithm, a threshold parameter, is optimized in order to maximize the sensor coverage and minimize the number of disconnected robots.**

I. INTRODUCTION

In recent years, the research on collective and swarm robotics has attracted significant attention of the roboticists who work on unstructured, unknown, and unpredictable environments. Originally inspired by the observation of social insects (ants, termites, wasps, etc.), swarm robotics deals with building collectively intelligent systems that are composed of large number of relatively simple physically embodied agents. The individual robots in the swarm are generally very simple in terms of computational, actuation, perception, and communication capabilities, however the overall and emergent behavior is complex.

Swarm-robotic systems are successful in environments where the prior-knowledge about the world is minimal and it is hard to build the model of the unstructured, dynamic, and unpredictable environment. Moreover, robotic swarms could be utilized in the environments where human intervention and robot-robot communication is very difficult if not impossible. Since the robots are usually identical, control of the swarm is distributed, and robots' behavior mostly depend on the local interactions with the environment, swarms could be utilized in risky and dangerous environments, exhibiting robust performance.

The objective of dispersion is to cover maximum area while maintaining the connectivity within the swarm. Dispersion of robotic swarms appears to be applicable and useful in domains such as planetary exploration, urban surveillance after a hurricane, and decontamination after a nuclear disaster, where the robots with maximal coverage collect samples from the unknown surface, detect the victims and collect the nuclear waste, respectively. The problem of dispersion could be related to the area coverage problem which is studied in individual [1] and collective level [2], [3] in depth. A large body of literature exists on the algorithms to maximize the covered area and find optimal placement of the robots. However, most of these algorithms require the global information and involve complex computations. Since the global information is not provided in our case and the robots are extremely simple (even cannot localize themselves), these solutions have little relevance within our framework.

In our framework, the robots with minimal communication and computational abilities are required to cover the maximum sensed area. Initially placed close to each other, the individuals in the swarm disperse based on local information gathered from surrounding robots and obstacles. In [4], robots in a virtual world are spread out by different movement algorithms such as random movement, wall-following, and driving towards open areas. Howard et.al. [5] applied a potential-field-based approach to the area coverage problem, where the robots (sensor nodes) are treated as virtual particles, and driven by virtual forces. The obstacles and other robots create a repulsion force if they are close and a viscous friction force is utilized to reach a state of static equilibrium. In [6], the simulated robots, which are equipped with two 2D laser range finders positioned back-to-back, also select a direction opposite to the dominant gathering of the other nearby robots and/or obstacles. All three studies mentioned above rely on the strong assumption that the robots are able to obtain the bearing and distance of the neighboring robots through their sensors. Since the robots are small, it is impossible to embed laser scanners in todays technology, as proposed in [5]. Although it is possible to used infrared sensors for this purpose, they have a very limited range. Moreover, in order to obtain a good estimate, one should place large number of infrared sensors on the robot, which corresponds to high power consumption.

Another alternative solution to estimate the distance from other robots is using intensity signals obtained from wireless communication and sensors. From a practical point of view, since wireless modules are small and power-effective, they are generally embedded in small-size robots. In [7], the range measurements obtained from wireless sensors are used in dispersion problem. Although, these sensors do not provide relative positions of the other robots, the swarm is able

1-4244-1364-8/07/$25.00 ©2007 IEEE

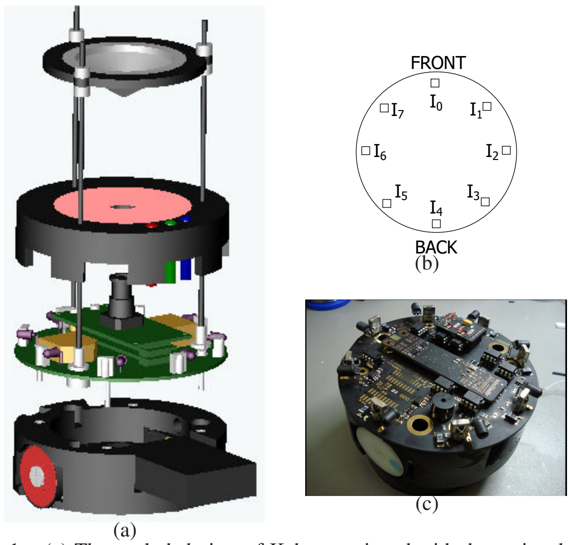


Fig. 1. (a) The exploded view of Kobot, equipped with the optional omni-directional vision system. (b) The positioning of the IR sensors, and their numbering. (c) The basic version of Kobot. The cap of the robot is removed, exposing the short-range sensing board and the main controller and the wireless communication boards stacked on top.

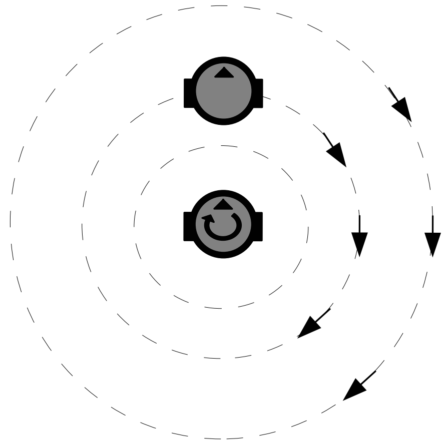


Fig. 2. Setup for experiments to measure the wireless intensity signals of the real robots in different distances and relative orientations. The robot in the center rotates around its own axis. The other robot is placed in different positions on the contours. In summary, (1) the orientation of the robot in the center, (2) the bearing of the other robot, and (3) the distance between them are changed in the experiments.

to disperse in different virtual environments based only on range information. Their work is based on the strong assumption that signal intensity decreases proportional with the square of the distance it travels. The intensity does not only depend on the distance between robots, but also depends on the structure of the antenna that is used and the surrounding environment.

In this paper, we will show that the signal intensity does not only depend on the distance between robots, but also is affected by the orientation of both robots. Moreover, the readings are very noisy and they largely depend on the environment characteristics. As a result, we employed a sampling technique [8] and a look-up table to model the sensor readings. Based on these readings, we utilized a very simple and flexible dispersion algorithm to disperse the robots in a simulated environment, while maintaining the connectivity.

The rest of the paper is as follows. First the robotic platform and its physics based simulator will be described giving a detailed overview of the wireless module. The control of the robot, which is designed based on the characteristics of the wireless sensor will be provided in Section III. Next, the results of the simulation experiments in which various numbers of robots are dispersed in different environments are provided. In the last section, the possible future directions are discussed.

II. Robot Platform

Kobot [9] is a circular differential-drive robot which is specifically designed to serve as a test-bed for swarm-robotic studies. Kobot (Figure 1) with a diameter of 120mm (the size of a CD) and a weight of 350 grams with batteries, is designed to be a light, small, yet extendable, power-efficient and relatively cheap robot platform for swarm robotics research.

The overall system design of Kobot is shown in Figure 1(a). At the heart of the Kobot, there is the control sub-system in which all of the information are fed from the other sub-systems, that is short-range sensing, communication, vision and power. Kobot comes with a novel IR-based short-range sensing sub-system shown in Figure 1(b) and 1(c). 8 infrared sensors are distributed around the robot, each of which is able to detect the objects in a half cone angle of 25° and $0.15m$ range. Kobot additionally provides wireless support using the IEEE802.15.4/ZigBee protocol. This protocol provides a low-power networking capability that can support point-to-point, point-to-multipoint and peer-to-peer communication.

The basic version of Kobot is planned to be extendable by a general-purpose omnidirectional vision sub-system as seen in Figure 1(a). This system is composed of a camera facing an omnidirectional mirror placed on top of the Kobot. It can view a region of $0.9m$ radius, shrinking the view with a constant proportion independent of the distance.

A physics-based simulator which is built on top of Open Dynamics Engine (ODE) is used in the experiments. The main body and wheels are modeled using basic cylindrical collision geometries. The actuators are simulated using virtual motorized hinge joints of ODE. Both the actuators and sensors are calibrated against the physical robot using the results of systematic experiments. For example, the virtual friction coefficients between wheels and ground, and virtual weights of the components are adjusted to obtain a similar movement pattern with similar motor torques. Infrared sensors are also calibrated using the data obtained from real sensor, and utilizing ray collision technique in the simulator.

A. Modeling of intensity signals

Since the dispersion method is based on the intensity signals obtained from wireless sensors, modeling of these sensors has crucial importance. The antennas of the wireless

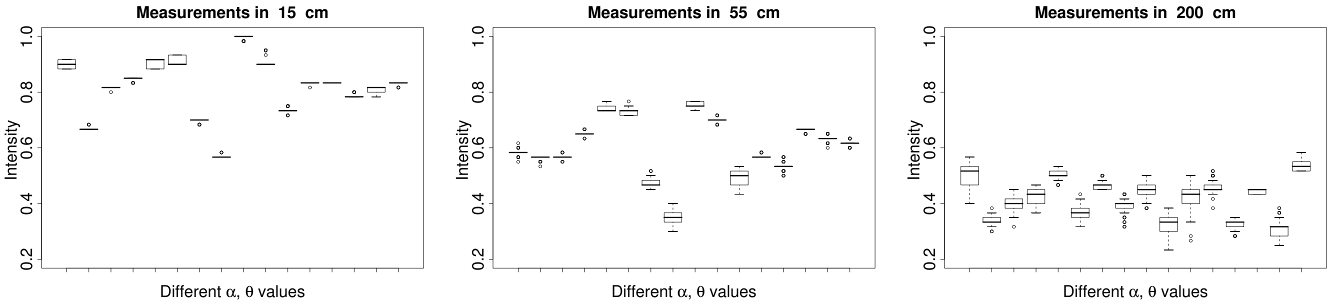


Fig. 4. The distribution of the intensity measurements are shown in detail. Each plot gives readings for different distances. Each box corresponds to a specific $\theta - r - \alpha$ triple, and represents the distribution of 200 intensity readings for a particular placement of the robots.

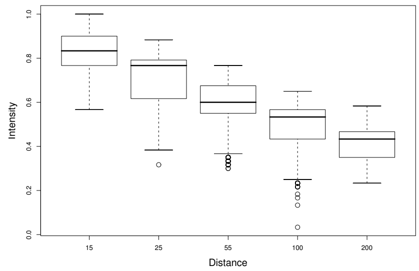


Fig. 3. The box-and-whisker plot shows the distribution of the intensities read from the wireless sensors for different distances between robots, each of which corresponds to a contour. The ends of the boxes and the horizontal line in it corresponds to second and third quartiles and the median points respectively. The outliers in the data are shown as circles in the plot.

modules do not have symmetric sensing characteristics, thus the relative orientations of the robots have unpredictable effects on the readings. We used two robots in our experiments, as depicted in Figure 2, one is in stationary position and rotating around its own axis, and the other is in different distances and relative positions with respect to the stationary one. The self-orientation angle of the stationary robot will be represented as θ, the distance and angle of the other robot will be represented as r and α respectively. In summary, the measurements are taken by changing 3 parameters, θ, r, and α.

In the experiments, the stationary robot is placed in 4 different orientations $\theta = \{0, \pi/2, \pi, 3\pi/2\}$. For each of the θ values, the other robot is placed in 5 different distances $r = \{15cm, 25cm, 55cm, 100cm, 200cm\}$ and in 4 different orientations $\alpha = \{0, \pi/2, \pi, 3\pi/2\}$. In each of these placements, 200 measurements are done. As a result, $(4 \times 5 \times 4) \times 200 = 16000$ wireless signal intensities are read in total. The box-and-whisker plot in Figure 3 shows the intensities obtained for different distances. Each box corresponds to the distribution of the measurements read in $4 \times 4 = 16$ different orientations. Although the intensity value decreases statistically with increasing distance between robots, the variances in the readings are very large. For example, the intensity value read in 15cm can also be obtained in 100cm.

Figure 4 shows the distribution of the intensity readings in detail. As shown, the noise in intensity readings become smaller when the robots are closer, and higher when robots are further. One important observation is that although the intensity does not decrease solely based on distance, it usually decreases with increasing distance for fixed α-θ values.

It appears to be difficult to model the characteristics of the sensor and the noise by fitting a function. As a result, the wireless sensor is modeled using the sampling data obtained from the real robots. Since it is possible to obtain the exact positions and orientations of the robots in the simulator; θ, r and α values are computed for each robot pair. θ and α are then rounded to one of the discrete values whose real samples exist. At the last step, an average intensity value is found for corresponding θ-α pair and closest two r discrete indexes.

III. Robot control algorithm

A subsumption like architecture is designed for the dispersion task. In the lower level, an obstacle avoidance behavior is executed when a close object is sensed by the infrared sensors which are located in front of the robot. If there is no close object in the frontal area, the robot executes dispersion behavior.

A. *Obstacle Avoidance*

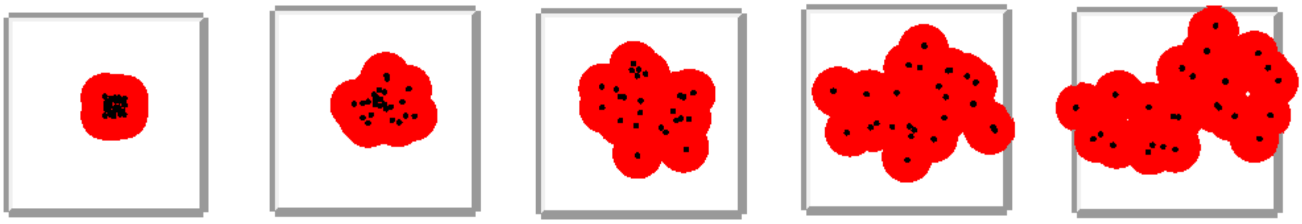
The robot while avoiding from obstacles and other robots, is controlled by setting speed of its left and right wheels (m_l and m_r), which are calculated as [10]:

$$m_l = (1 - |\bar{r}|) * 0.25 - \bar{r}$$
$$m_r = (1 - |\bar{r}|) * 0.25 + \bar{r}.$$

where $\bar{r}$ denotes the tendency to turn. When $\bar{r} = 0$, the robot moves forward. It turns left when $\bar{r} = 1$, and right when $\bar{r} = -1$. Here, $\bar{r}$ is defined as $sign(w_r - w_l) * \bar{n}$, where n is a random number between -0.4 and 0.4, $\bar{n}$ is a random number between 0.3 and 1.0, w_l, w_r represent the 'perceived presence' of the wall on the right and left side respectively, r is defined as the value of the 'rotational activation'. In this formulation, the robot would make a turn of random size in its current turning direction.

The change in r is calculated as

$$\begin{aligned}\Delta r = \;& -0.9r \qquad . \\ & +0.3(1-r)(w_l + 1.5I_1 + 1.2I_0) \\ & -0.3(1+r)(w_r + 1.5I_1 + 1.2I_7)\end{aligned}$$



(a) Initial (b) 100 steps (c) 250 steps (d) 500 steps (e) 1000 steps

Fig. 5. A swarm of robots of size 25 are dispersing in a $100m^2$ square shaped room in the physics based simulator. The black small points show the positions of the individual robots, and red circles represent the sensor coverage for each robot. The snapshots are taken in different timesteps. (a) shows the initial placements and initial sensor coverage of the robots. In (e) after 1000 simulation time steps, approximately half of the room is covered by the sensors of the robots.

The first term on the right of the equation guarantees that when no wall is perceived and the infrared readings are all zero, then any rotational activation will decay to zero in time. The second term raises the rotational activation towards 1 in proportion to the amount of wall perceived on the left side and the infrared readings from the right side. The third term tries to pull down the rotational activation to -1 in a similar way. I_i denotes the infrared readings, with a value between 0 (no object) and 1 (very close object), where $0 < i < 8$ is the index (Figure 1(b)).

The variables, w_l and w_r, indicate the presence of the peripheral wall on the left and right side of the robot respectively, and the change in them are defined as

$$\Delta w_l = -0.1 w_l - 0.7 w_l (I_7 + I_1)$$
$$\Delta w_r = -0.1 w_r - 0.7 w_r (I_7 + I_0).$$

The first term on the left side causes the perceived presence of a wall to decay to zero when no objects are sensed. The second term diminishes the perceived presence of any wall if the front sensors become active, to raise the priority of avoidance. Even with obstacle avoidance in place, the robot can get stuck, particularly when it is moving straight towards the wall. The first condition of $\bar{r}$ allows robot to escape from such situations by making steep turns away from the obstacles blocking its course of movement.

B. Dispersion

Similar to the potential-field based methods, the robot is repelled by other robots when they are closer than a threshold, and is attracted by them when they are further than that threshold. While the first "force" applied to the robot make the dispersion possible, the latter one ensures the connectivity of the swarm.

Two characteristics of the readings obtained from wireless sensors make the utilization of the attraction-repulsion idea difficult. First, in potential-field-like methods the bearing information of the surrounding robots are required to find the direction of each of them, and compute an average direction vector towards or away from them. Since the wireless sensors do not provide the bearing information, the vectorial approach could not be applied directly. The other problem with wireless sensors is that they are very noisy as discussed in the previous section. A method, that is very robust to the noise should be utilized.

Our method is designed based on the following idea: Assume that the robot moves in a certain direction. If the wireless intensity decreases during its move, it can be deduced that the robot is moving away from other robots. If it increases, the robot is most probably approaching to other robots. Although the intensity readings differ in large magnitude for robots in same distances but in different relative orientations, they are observed to decrease for same orientations and increasing distances. Thus, if it is assumed that the change in relative orientations is very small during the motion of the robots, the inverse proportional relation between wireless signal intensity and distance between robots would hold.

The speeds of the left and right wheels are set as:

$$m_l = \begin{cases} V_f + r & : \quad W < \tau \wedge \Delta W > 0 \\ V_l + r & : \quad W < \tau \wedge \Delta W < 0 \\ V_l + r & : \quad W > \tau \wedge \Delta W > 0 \\ V_f + r & : \quad W > \tau \wedge \Delta W < 0 \end{cases}$$

where r is a random real number, W is the intensity of the wireless signal, ΔW is the difference between current and previous intensity readings, and τ is the threshold to determine whether escape from robots and enable swarm dispersion or move towards robots and maintain connectivity. The formula of m_r is same as m_l except V_r replaced V_l. V_f is used for forward movement, and $V_l < V_r$ enables the rotation of the robot. Since the wireless readings are highly noisy, the intensity value is computed using the previous readings, ie. $W = 0.8 \times W_t + 0.2 \times (0.8 \times W_{t-1} + 0.2 \times ..)$.

When the current intensity is bigger than the threshold $\tau = 0.4$ and the intensity value is decreasing, it means that the robot is moving away from other robots. In this case, both wheels are set to approximately to the same speeds, enabling the robot to go forward and continue escaping from others. However when the intensity value is increasing, it means that the robot moves towards others and it should change its route to enable dispersion. If the current intensity is smaller than 0.4, the robot should return back instead of escaping from others in order not to lose the connectivity.

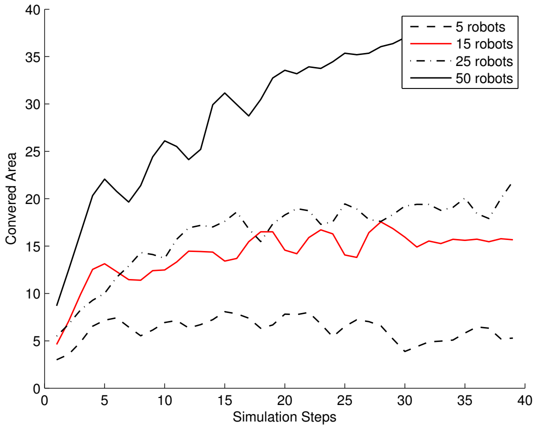


Fig. 6. The change of the sensor area coverage in a $50m^2$ room for various swarm sizes.

IV. Experiments

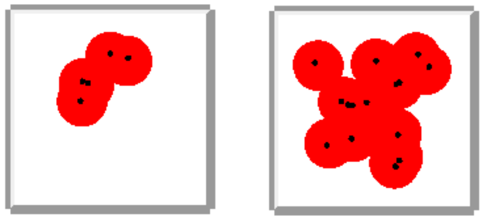
The experiments are conducted in the physics based simulator described in Section II. In all experiments, the robots are placed at the center of the room with random orientations, and the dispersion algorithm is run for 2000 simulation steps. Figure 5 shows a sample run for 10 robots and a $100m^2$ room. Around each robot, a red circle is drawn, which demonstrates the robot's sensor coverage range. The range of the omni-directional camera (Figure 1(a)), which is $0.9m$, is set as the radius of this circle.

The effect of the number of robots in the swarm is examined in another experiment. 5, 15, 25, and 50 robots are placed in a $50m^2$ room, and the algorithm is run for 2000 simulation steps. The areas covered by the robot sensors are measured in each time step and demonstrated in Figure 6. As shown, the covered area increases with increasing number of robots. Although the room is shown to be covered with sufficient number of robots, two problems appear in the plots. First, while optimally dispersed 20-30 robots can cover the room completely, even 50 robots are not able to cover the room in our experiments. Second, oscillations are identified in the plots, which are the result of slowly dispersing swarm behavior where the group periodically expands and shrinks. In summary, our method could not reach optimal solution and has a slow convergence rate.

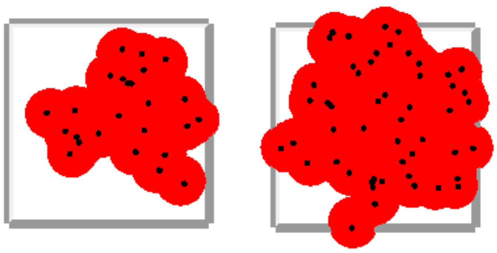
The snapshots of the swarms after 2000 steps are shown in Figure 7. As shown, the swarm generally remains connected, despite no global criteria is set to ensure this constraint. Although during dispersion, some of the robots are disconnected from the swarm, the circular movement paths of the robots enable them to find and reconnect to the others.

A. The effect of threshold τ parameter

The parameter τ determines the distance threshold where the state transition from robot repulsion to robot attraction occurs. For big τ values, the swarm should be able to cover larger areas, but have a larger disconnection probability at the same time. On the contrary, if τ is smaller, the swarm remain connected but could not disperse effectively. In this experiment, we examined the effect of τ parameter by changing τ between $[0-1]$ in 5 discrete steps. In order to allow situations where robots could be disconnected from the swarm, a big room ($200m^2$) and 15 robots are used. Figure 8 shows the change in covered area based on the threshold parameter. The largest covered area is obtained for $\tau = 1.0$ which corresponds no attraction force. Since the robots do not preserve the connectivity, the area that is sensed by the whole swarm is maximized. Figure 9 shows the snapshots of the robots and sensor coverage at the step 2000. While the robots with small τ values are almost compact, the robots with big τ values are completely disconnected. The optimum value of τ is around 0.5.



(a) 5 robots (b) 15 robots



(c) 25 robots (d) 50 robots

Fig. 7. The sensor area coverage is shown for various swarm sizes. The snapshots are taken at the 200^{th} steps.

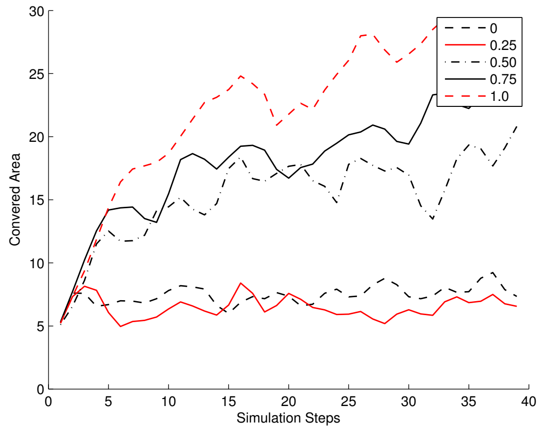
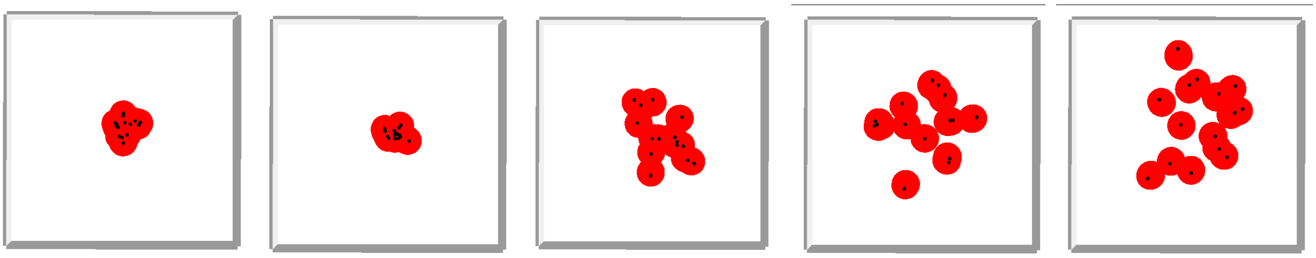


Fig. 8. The change of the sensor area coverage in a $200m^2$ room for different τ values.



(a) $\tau = 0$ (b) $\tau = 0.25$ (c) $\tau = 0.50$ (d) $\tau = 0.75$ (e) $\tau = 1.00$

Fig. 9. The sensor area coverage is shown for different τ values. 25 robots are deployed in a virtual room of area $200m^2$. All snapshots are taken at 2000^{th} simulation step. While small threshold values result in compact swarms, big threshold values result in unconnected dispersed swarms.

V. CONCLUSION

In this paper, a method for the dispersion of a robotic swarm is proposed, where the intensity readings obtained from wireless sensors are used as range estimates of the surrounding robots. Since the experiments are conducted in a physics based simulator of the particular robot platform, the modeling of the wireless sensors has crucial importance. In the experiments performed with real robots, it is observed that the signal strength is highly affected from the orientations of both communicating robots. As a result, systematic experiments on real robots are performed to sample the intensity signals based on distance between robots and relative orientations. The readings that are obtained in these experiments are then transferred to the simulator and a look-up table is utilized in calculating the virtual intensity values.

A simple algorithm, which is similar to the potential-field based approaches is proposed in the paper. Although the bearing of other robots could not be obtained from wireless sensor readings, and the noise is very high (and unpredictable most of the time), this algorithm successfully disperses the robots in environments surrounded by walls. The only parameter of the algorithm, a threshold parameter, is optimized in order to maximize the sensor coverage and minimize the number of disconnected robots. When the threshold is adjusted as a small value, connected but compact swarms are obtained. On the contrary for big threshold values, the swarm is able to disperse in the environment, however connections between the robots loosen up.

This study could be extended in many different ways. The antenna used in the wireless sensor is not symmetric and this structure makes the modeling of the readings very difficult. Changing the antenna with a symmetric one would ease and speed up the sampling process because the readings become independent of the relative orientations of the robots. Better controllers which depend on the simpler wireless models could be designed in this way.

The main aim of this paper was to study wireless intensity signals and dispersion problem in a realistic framework. The next steps are to improve the dispersion algorithm, test it in more realistic environments such as office environments, and then transfer it to the real robots.

VI. ACKNOWLEDGMENTS

This work is funded by TUBITAK 104E066 research grant.

REFERENCES

[1] I. A. Wagner, M. Lindenbaum, and A. M. Bruckstein, "Mac versus pc: Determinism and randomness as complementary approaches to robotic exploration of continuous unknown domains.," *I. J. Robotic Res.*, vol. 19, no. 1, pp. 12–31, 2000.

[2] H. Choset, "Coverage for robotics - a survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, pp. 113 – 126, 2001.

[3] D. Gage, "Command control for many-robot systems," in *Proceedings of the Nineteenth Annual AUVS Technical Symposium*, (Huntsville, Alabama), pp. 28–34, June 1992.

[4] R. Morlok and M. Gini, "Dispersing robots in an unknown environment," in *Proceedings of International Symposium on Distributed Autonomous Robotic Systems*, (Mineapolis, Minnesota), pp. 241–249, July 2006.

[5] A. Howard, M. J. Matarić, and G. S. Sukhatme, "An incremental deployment algorithm for mobile robot teams," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (Lausanne, Switzerland), pp. 2849–2854, Oct 2002.

[6] M. Batalin and G. S. Sukhatme, "Spreading out: A local approach to multi-robot coverage," in *Proceedings of the International Symposium on Distributed Autonomous Robotic Systems*, (Fukuoka, Japan), pp. 373–382, Jun 2002.

[7] L. Ludwig and M. Gini, "Robotic swarm dispersion using wireless intensity signals," in *Proceedings of the International Symposium on Distributed Autonomous Robotic Systems*, (Mineapolis, Minnesota), pp. 28–34, July 2006.

[8] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo, "Swarm-bot: A new distributed robotic concept," *Auton. Robots*, vol. 17, no. 2-3, pp. 193–221, 2004.

[9] http://ko-bot.blogspot.com/.

[10] Sahin, Erol, Girgin, Sertan, Ugur, and Emre, "Area measurement of large closed regions with a mobile robot," *Autonomous Robots*, vol. 21, pp. 255–266, November 2006.