Single-Sensor Probabilistic Localization on the SeReS Self-Reconfigurable Robot

Kenneth PAYNE, Jacob EVERIST, Feili HOU and Wei-Min SHEN[1]
University of Southern California, Information Sciences Institute, USA

Abstract. This paper proposes a novel method for localizing a stationary infrared source of unknown orientation relative to a static docking sensor. This method uses elliptical approximations of likely positions of the infrared source and computes the intersections to find the most probable locations. It takes only a few samples to localize, is easily computed with inexpensive microcontrollers, and is robust to sensor noise. We then compare our approach with two other methods. The first uses a Bayesian filter across a map of discrete locations in the robot's operational workspace to determine the suspected source position. The second also uses a probability distribution map but uses the method described by Elfes in his paper on probabilistic sonar-based mapping and navigation [1]. We show that our approach localizes quickly with a single sensor and is no more computationally demanding than other methods.

Keywords. Alignment, Docking, Bayesian, Infrared Beacon, Elliptical Probabilistic Sensing, SeReS, CONRO.

Introduction and Previous Work

Self-reconfigurable robots are modular systems that can change and adapt their configuration to the environment based on the needs and characteristics of their task. These robots rely heavily on their inherent ability to connect and disconnect multiple times to change the configuration of modules in the network. A system that is able to align and correct for docking offsets can change shape reliably.

One of the fundamental problems encountered during reconfiguration is the issue of correcting the error in connector alignment. Traditionally, the reconfiguration process would be hand-coded into the system by either an operator or choreographed by an off-board computer [2]. However, one of the goals of self-reconfigurable robots is to achieve reconfiguration at run-time [3][4][5][6][7]. To accomplish this, a method of localization is needed to overcome noise in motors and proprioceptive sensors, as

[1] Corresponding Author: Wei-Min Shen, USC/Information Sciences Institute, 4676 Admiralty Way, Suite 1001, Marina del Rey, CA 90292, USA; E-mail: shen@isi.edu.

well as error caused by docking friction and slippage. Many researchers have addressed this issue for both reconfigurable and general robotic docking [8][9][10][11][12]. Occasionally, those phenomena are somewhat predictable and can be approximated. For single-sensor modular systems, alignment with the infrared beacon was typically accomplished by scanning back and forth, localizing through multiple passes [10][13]. Systems with multiple sensors usually involved large amounts of computational complexity [14]. Lastly, the discretized methods discussed are based on the Bayes filter as well as Elfes' work on sonar range finding [1].

Probabilistic methods achieve their objectives through sparse sampling and computation of probability distributions, which are robust to sensor and actuator noise. We designed our probabilistic approach for detecting error in sensor alignment and compared it to two others. For this series of experiments, we focus on the detection and localization of a single infrared light. The novel method we have developed is a parametric approximation of the two probability distribution-based methods. If we derive an accurate sensor model, the accuracy of the probabilistic methods is not significantly affected by sensor noise. The advantage of our approach over others is that it localizes quickly with a single sensor, yet uses no more computation than other methods, and can be carried out on a simple, inexpensive on-board microcontroller.

1. Experiment Characteristics

1.1. CONRO SeReS Robot Module and Testing Environment

CONRO is a self-reconfigurable robot system (Fig. 1a). Each module has four interconnecting dock faces with an infrared emitter/receiver pair (for digital communication and analog sensing) as well as a pair of pin-and-latch docking systems, discussed in [12]. For this experiment, we used a CONRO SeReS module equipped with an Atmel Atmega128L processor[2] and 128K flash memory.

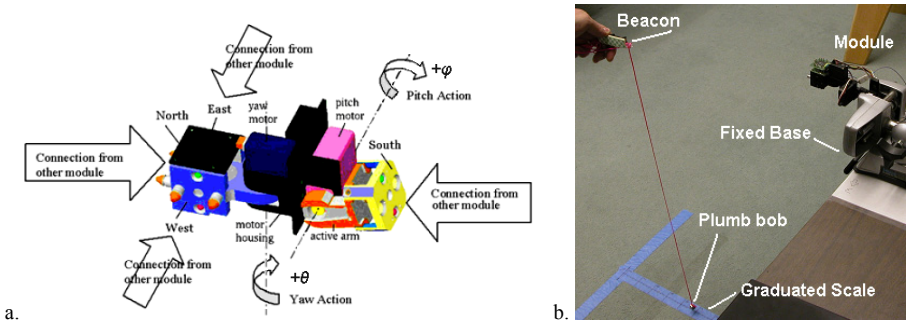

Figure 1. (a) The CONRO Self-Reconfigurable Robot. (b) The SeReS conversion test environment. The operator holds a mobile infrared beacon with a plumb bob above a graduated strip mounted below the robot.

All three experiments were conducted in the same room (in Figure 1b) under the controlled lighting conditions of a typical office environment with diffuse fluorescent lighting in the ceiling, matte white walls, and celadon carpeting. This is by no means an approximation of the real world, but is instead used as an initial testing environment

[2] For more information on this processor, see http://www.atmel.com, and also http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf.

for the proof-of-concept work done in this experiment. Further testing should involve variable light and reflectivity in the environment, such as under direct sunlight, and ambient or cloudy skies.

1.2. Sensor Models

For our sensor model, we simply use an approximation of the real light characteristics of the infrared detector on the SeReS module. SeReS reads in a 10-bit brightness value that is then used to determine a suspected angular offset for both yaw θ and pitch ϕ (Figure 2). We collected sensor response data in 100° vertical and horizontal sweeps to map detector sensitivity to both rotational axes. This was a simple yet appropriate approach since SeReS modules often perform side-to-side or up-and-down motions to find their neighbors in reconfiguration tasks.

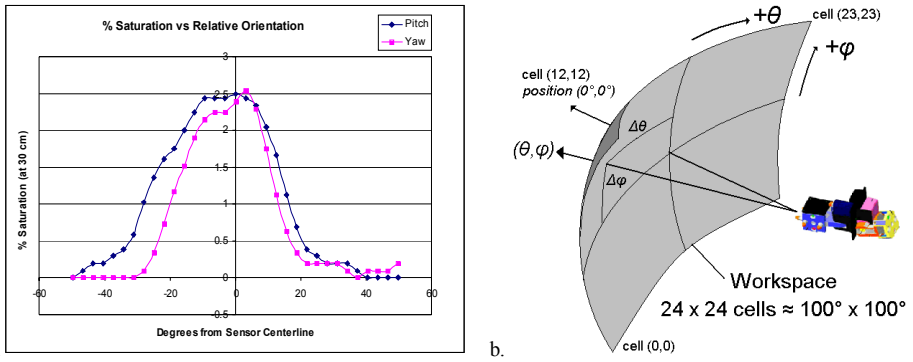

Figure 2. Brightness vs. Angular Offset Sensor Model. Note that in (a) "% Saturation" refers to total sensor saturation. Since these experiments were conducted at 30 cm, a very small but sensitive range is detected, and the infrared detector never completely saturates. This is from a 10-bit value representing a 5V range. (b) shows how the orientation workspace of SeReS is mapped onto a 2D spherical section. Note also that orientation can be expressed in degrees (0°,0° center) or in discretized 4.2° x 4.2° cells (12,12 center cell).

2. The Elliptical Intersection Method

The most important part of a probability distribution map is the set of peaks indicating likely regions in which the source should be found. As the sensor model can only approximate range but not specific orientation from the current position, each location has with it an associated elliptical region of likely positions. The elliptical intersection method represents the highest rings of those regions of probability as parametric ellipses. By determining the intersections of these ellipses (Figure 3), a concise set of points can be found where the infrared source may be located. In our light model, we measured the response curves of the infrared detector on both θ and ϕ axes. This simple model ensures that all major and minor axes of associated ellipses will be parallel. The major and minor axes of each ellipse are the hypothesized angular offset taken from the light reading in Figure 2a. Lastly, the three samples are arranged in a triangle to cover more operational space.

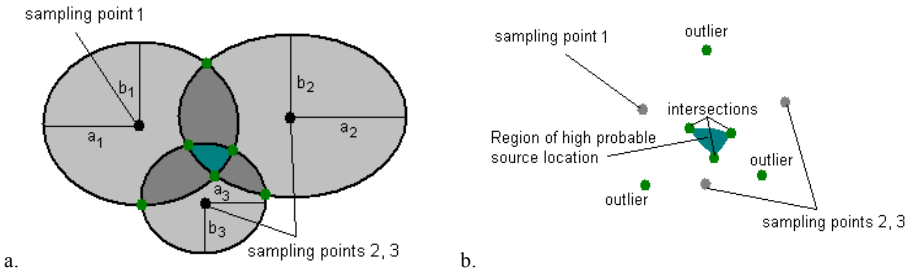

Figure 3: (a) The elliptical intersection method begins by calculating the sizes of three sampling ellipses, taken from different points in SeReS' operational space, and whose *a* and *b* terms are determined from the light measured at each sample point with respect to the light model from Figure 2a. (b) shows the resulting intersections of the ellipses determined in (a). After discarding the outliers, the remaining cluster will envelope the region most likely to contain the infrared source.

Recall from elementary geometry that the intersection of two ellipses is Eq. (1) and Eq. (2) after simplification. In these equations, the variables *a* and *b* are the vertical and horizontal angular offsets taken from the light model in Figure 2a, and the position (*h*, *k*) is each point at which sampling takes place.

$$\frac{(\theta - h_1)^2}{a_1^2} + \frac{(\phi - k_1)^2}{b_1^2} = \frac{(\theta - h_2)^2}{a_2^2} + \frac{(\phi - k_2)^2}{b_2^2} \tag{1}$$

$$\left(\frac{1}{a_1^2}\right)\theta^2 - \left(\frac{2h_1}{a_1^2}\right)\theta + \left(\frac{h_1^2}{a_1^2} + \frac{(\phi - k_1)^2}{b_1^2}\right) = \left(\frac{1}{a_2^2}\right)\theta^2 - \left(\frac{2h_2}{a_2^2}\right)\theta + \left(\frac{h_2^2}{a_2^2} + \frac{(\phi - k_2)^2}{b_2^2}\right) \tag{2}$$

As the current SeReS orientation (*h*, *k)* and the suspected angular offsets *(a, b)* will not be known until sampling at runtime, a general solution is needed. We used Sylvester's determinant of the matrix Eq. (3) to remove the θ term. The determinant itself is a very long fourth order polynomial with thirty-four terms. This allows us to find its ϕ roots by simplifying it into the form of Eq. (4) and solving.

$$\begin{bmatrix} \frac{1}{a_1^2} & -\frac{2h_1}{a_1^2} & \frac{h_1^2}{a_1^2} + \frac{(\phi - k_1)^2}{b_1^2} & 0 \\ 0 & \frac{1}{a_1^2} & -\frac{2h_1}{a_1^2} & \frac{h_1^2}{a_1^2} + \frac{(\phi - k_1)^2}{b_1^2} \\ \frac{1}{a_2^2} & -\frac{2h_2}{a_2^2} & \frac{h_2^2}{a_2^2} + \frac{(\phi - k_2)^2}{b_2^2} & 0 \\ 0 & \frac{1}{a_2^2} & -\frac{2h_2}{a_2^2} & \frac{h_2^2}{a_2^2} + \frac{(\phi - k_2)^2}{b_2^2} \end{bmatrix} \tag{3}$$

$$\frac{\left(A\phi^4 + B\phi^3 + C\phi^2 + D\phi + E\right)}{F} = 0 \tag{4}$$

Since the intersections are represented by floating point numbers, we must use the Newton-Raphson method of approximating roots the same way it is done on modern graphing calculators. By solving Eq. (1) for θ and substituting the roots found, one can find the set of real 2D points at which the ellipses intersect.

Next, by clustering the points together, we approximate the location of the source by discarding outliers and taking the center of mass of the remaining cluster. The sensor model guarantees that the major and minor axes of the ellipses will be parallel to the pitch and yaw axes of the SeReS module respectively. Also, the arrangement of sample points chosen for taking readings guarantee that there will be up to nine intersections. To simplify clustering, simply discard those points whose angular offset from the center of mass is greater than the average, and then recalculate the new center of mass. Due to the guarantee of axial symmetry and triangular arrangement, this clustering method is sufficient. If the light model were more complex, we would likely have to draw a clustering method from [15]. It should be noted that if fewer than three intersections are found, the data is insufficient to produce a solution. This is addressed later.

3. Discretized Methods

In these localization methods, we separated the angular offset space into a set of discrete cells that contain the probability of the light source being in that location. We used a 24 x 24 2D map of the SeReS module's yaw θ and pitch ϕ operational space that is also interpreted as the orientation of the infrared detector. This map places cell (0, 0) at the lower west corner and (23, 23) is at the upper east (Figure 2b). To update the belief functions, readings must be taken from various locations. This yields a ring or peak of high probability around SeReS cell location depending on the infrared source's suspected location. If the new probability for a given cell in the map is lower than the current belief for that position, it will decrease. This eliminates competing peaks from the map. We chose the four cells (12, 12), (5, 12), (14, 6), and (14, 14), because together these locations cover SeReS' entire operational workspace.

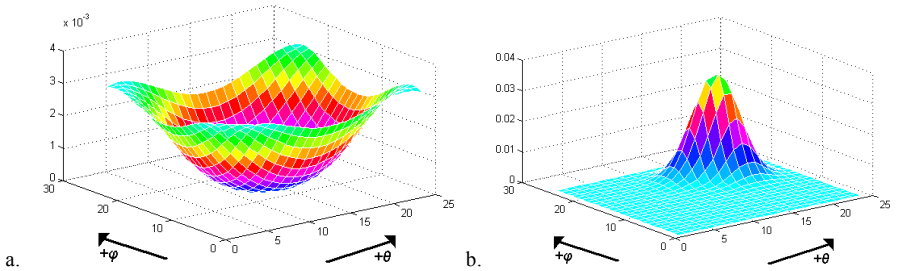

Figure 4: In both images, readings are taken from SeReS' central orientation at cell (12,12). In (a), the suspected brightness is far away from cell (12, 12) so an elliptical valley is created in the probability distribution. In (b) the source is very close to SeReS' current orientation, so a very small ring (becomes a peak) is created around the current orientation of the SeReS module.

For all sensed values b > 4.3% of complete infrared detector saturation, the function would return suspected angular offsets of 0°. This occurs mainly at perfect

alignment, and supports the assumption that the suspected source position was at an angular offset of 0° from SeReS' current position. This also creates a peak at the current position in the 24 x 24 map. The next part of the curve shows that for all brightness values 0.1% < b < 4.3%, the curve can be approximated by a parabola opening downwards. Lastly, for all sensed brightness b < 0.1%, a wide distribution was used for each cell. This essentially lowers the probability of the region surrounding SeReS and leaves the oblique regions with a nearly equal chance of containing the source. Not every point is expected to see the light source, and so this method maintains probability mass for those points not in the immediate detector range. See Figure 4 for illustration.

3.1. Implementation of Bayesian Filter

Recall the Bayes filter, governed by Eq. (5) for sensor observations. The robot starts with a uniform prior belief distribution of the light source. The state, s, is the position of the sensor in the 24 x 24 cell map. Due to the large size of the individual cells, (4.2° x 4.2°), Bayes' action model tested to be $P(s \mid s', action) = 1.0$ for all commanded servo motions, and is therefore left out for brevity. In Eq. (6) we represent the state as the coordinate pair (θ, ϕ). The observation, o, in Eq. (5) is the detected brightness B and the suspected angular offsets (a,b) that follow from the light model given in Figure 2a. The evidence η is found in the usual way by taking the summation of all non-normalized P and then dividing each cell by that value.

$$Bel'(s \mid o) = \eta P(o \mid s) * Bel(s), \text{ where } \eta = \frac{1}{evidence}. \tag{5}$$

$$Bel'(\theta, \phi \mid B, a, b) = \eta P(B, a, b \mid \theta, \phi) * P(\theta, \phi) \tag{6}$$

3.2. Implementation of Elfes' Method

In pursuit of another method for updating the prior, we also tried Elfes' approach [1]. Elfes held that the new belief of a cell in an observation is governed by Eq. (7).

$$P(cell) = P(cell) + P(reading) - P(cell) * P(reading) \tag{7}$$

We picked this method because sound and light dissipate in a similar fashion, and we hypothesized that what worked for Elfes' on sonar-range finding would work on beacon location. Our infrared detector gives us magnitude but not direction just as Elfes' sonar gave magnitude (and by virtue of positioning on the robot, direction), but did so with heavy noise [1]. As Elfes used repetition in his method to circumvent the sonar noise, we used it to overcome the noise in the infrared detector. The rest of the process is carried out in the same fashion as for our Bayesian method.

4. Findings

4.1. Elliptical Intersection Method

For each method, we ran ten trials with the infrared source placed in various locations. For the elliptical intersection method, the results were erratic. In simulation, the system used very regular curves for brightness and always measured perfectly. In real application however, the sample points sometimes yielded 0% intensity, leaving us with the challenge of representing an "out-of-range" condition in the program. We tried two possible sets of angular offsets for this condition. The first was the bare minimum angular offset for that received value, which was a = 45°, b = 36°, taken from the model (Figure 2a). The second pair was the absolute extreme angular offset measured for 0% intensity: a = 50°, b = 50°. Some of the results are in Table 1.

Table 1. Out-of-Range Elliptical Intersection trial results.

Trial	Actual Position (θ, φ)	Cluster Center (θ, φ)	Out of Range Angular Offsets
EIM Trial I-a	(17°, 17°) = cell (16, 16)	(15°, 24°) (11°, 16°)	*a = 50°, b = 50°* *a = 45°, b = 36°*
EIM Trial I-b	(0°, 0°) = cell (12, 12)	(2°, -3°)	Both conditions.
EIM Trial I-c	(-50°,-50°) = cell (0, 0)	(7°, -8°) (-6°, -8°)	*a = 50°, b = 50°* *a = 45°, b = 36°*
EIM Trial I-d	(-30°, 25°) = cell (5, 18)	(-33°, -30°)	*a = 50°, b = 50°*

In theory, this method was able to approximate the light source in simulation to within ±2°. In actual practice, however, this was usually greater than ±10° with occasional trials such as I-d, off by ±60° on one axis. This indicates that our light source and light model were not as uniform as we had hoped, and that further study into making a clean reliable uniform light source will be needed. One benefit of the probabilistic methods was that even if no brightness was measured, the algorithm would increase the extreme areas and decrease the probability mass for the surrounding area. With the elliptical method, no reliable ellipse dimensions could be determined from a single measurement.

Lastly, the Newton-Raphson part of the algorithm had to cycle many more thousands of times for each potential intersection, which yielded a solution on our 4MHz processor within 10 to 30 seconds, (depending on the points selected and their respective ellipse sizes). This is currently too slow for implementation on real-time docking problems.

4.2. Discretized Methods

The discretized methods worked much more smoothly. The results in Table 2 show the discretized algorithms to be accurate usually within 2 cells as the probability increases. The Bayesian method is quick and accurate. It repeatedly localized to a single peak in short order; always within a few cells from the actual source of infrared light (it tends to converge low-west of actual position). Of particular interest were the results of Trial B-a, where the light source was placed slightly east and low-of-center. This was able to

converge on the fourth sample despite the fact that the first three measurements were out of range (Figure 5).

Table 2. Notable Bayesian and Elfes trial results.

Trial	Actual Cell (θ, φ)	Peak Cell (θ, φ)
Bayesian Trial B-a	(16, 16)	(18, 17)
Bayesian Trial B-b	(12, 12)	(14, 14)
Bayesian Trial B-c	(12, 12)	(16, 16)
Bayesian Trial B-d	(16,16)	(11,15)
Elfes' Method Trial E-a	(12, 12)	(1, 10)
Elfes' Method Trial E-b	(0, 0)	(3, 1)

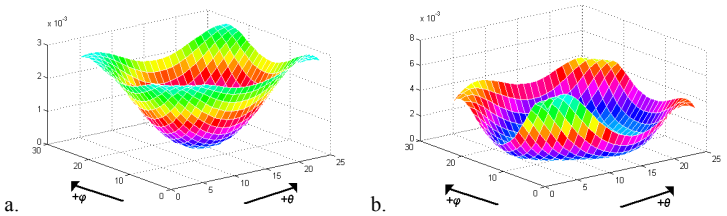

Figure 5: Trial B-a. (a) SeReS starts at cell (12,12) and sees no light. It then moves to cells (5,12) and (14,6) with no detected light. Finally, it approaches (14,14) (b) near the source and begins to converge with a higher peak. Already by the fourth sample (b), the three other competing peaks were reduced.

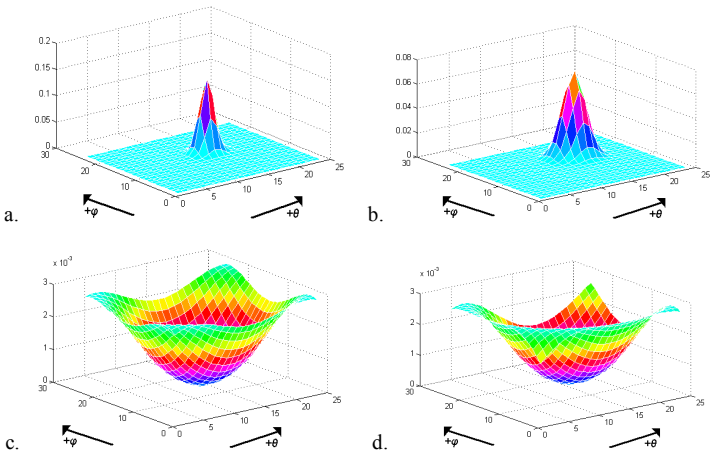

Figure 6. Trials B-b (a) and B-c (b) show the effects of noise in the sensor. Trial B-b has half of Trial B-c's standard deviation. The noise in the sensor yielded different probabilities, yet still converged to within 4 cells. In Trial E-a, the distribution does not readily converge. (c) SeReS begins at cell (12, 12). (d) Fourth step at (14, 14). The overall peak increased and moved closer to the source with each sampling, but a large portion of the probability ring still remains. To implement Elfes' approach more efficiently, we may need a better sensor model, or many more readings.

Bayesian Trials B-b and B-c, and Elfes' Method Trial E-a were all taken with the light at the center. Trial B-b registered the peak closer to the actual position, and Trial B-c had a wider standard deviation (Figure 6a and 6b). Next is trial E-a, which had the light source at the center (12, 12). The adaptation of Elfes' approach is slow, but promising. To effect change in the probability distribution it relies on many readings, and had not yet converged to a single useful peak by the fourth sample. It is shown in Figure 6c and 6d.

5. Conclusions and Future Work

One conclusion we can draw from these results is that the chief cause of failure or inaccuracy was that our sensor model was too brief, and did not account for anomalies that were seen at oblique angles from above and below. These anomalies are common to off-the-shelf infrared detectors like the ones used on SeReS, and are even within the specifications of the particular one we used. This is supported by the fact that the simulated models worked well based on simplified physical laws that govern light behavior. Once the methods were applied under real-world conditions, the shortcomings of the light model became apparent.

Also, we found that by reducing the standard deviation for the sensor model probability curve, the peaks were sharper and converged faster and tighter than when we used a broader curve. This is only natural, as more probability mass was concentrated into a smaller volume. This accounts for the difference in Bayesian Trials B-b and B-c. By giving more benefit of the doubt to nearby cells, Trial B-c converged with a wider distribution around the peak, whereas Trial B-b was higher and tight.

While this paper focused on localizing a single infrared source under semi-ideal conditions, it should be noted that the real world is full of light noise and sensor misalignments. It is therefore desirable to develop a more rigid and redundant sensor model that takes into account other factors such as slop and backlash in the servomotors, multiple light sources, and of most importance, moving lights at various ranges from the detector. One suggestion we have for further work on this topic is to use a weighted system of constant random sampling. The peaks or intersections would quickly drift across the operational space and the clusters would track the source with an error whose size was directly related to the sampling frequency and source velocity.

Another suggestion would be to use a variable set of light models taken at varying distances. If peaks or intersections are either very far away from each other, or non-existent, then the light model would be assuming that the source is too close, or too far away, respectively. One could use a wide variety of standard approaches to converge on selecting the proper model, such as rule-based lookup tables, simulated annealing, neural networks, and many others.

A proper infrared sensor model can help modular systems to localize relative to opposing dock faces during self-reconfiguration. Using probabilistic methods, the approach phase of docking can be sloppier and still localize and align, assuming the control algorithm chooses appropriate points from which to take readings. In our experiments, points were chosen arbitrarily for apparent uniform distribution across the operational space, but a more variable approach could also be used.

Acknowledgements

We are grateful that this research is sponsored by AFOSR under award numbers F49620-01-1-0020 and F49620-01-1-0441. We also want to thank other members in the Polymorphic Robotics Laboratory for their generous intellectual support and discussions.

References

[1] A. Elfes, "*Sonar-based Real-world Mapping and Navigation",* IEEE Transactions on Robotics and Automation, 3(4), pps. 249-265, 1987.

[2] E. Yoshida, S. Murata, A. Kamimura, K. Tomita, H. Kurokawa, S. Kokaji, *"A Self-Reconfigurable Modular Robot: Reconfiguration Planning and Experiments", International Journal of Robotics Research,* 21(10), pps. 903-916, October 2002.

[3] M. Rubenstein, M. Krivokon, W.-M. Shen, "Robotic Enzyme-Based Autonomous Self-Replication", International Conference on Intelligent and Robotic Systems, Sendai, Japan, 2004.

[4] Z. Butler, K. Kotay, D. Rus, K. Tomita, "Generic Decentralized Control for a Class of Self-Reconfigurable Robots", Proceedings of International Conference on Robotics and Automation, Washington, DC, USA, 2002.

[5] K. Stoy, R. Nagpal, "Self-Repair Through Scale Independent Self-Reconfiguration", Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, 2004.

[6] K. Kotay, D. Rus, "Generic Distributed Assembly and Repair Algorithms for Self-Reconfiguring Robots", Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, 2004.

[7] E. Klavins, R. Ghrist, D. Lipsky, "Graph Grammars for Self-Assembling Robotic Systems", *Proceedings of International Conference on Robotics and Automation*, New Orleans, USA, 2004.

[8] M.W. Jorgensen, E.H. Ostergaard, H.H.Lund, "Modular ATRON: Modules for a Self-Reconfigurable Robot", Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, 2004.

[9] M. Nilsson, "Heavy-Duty Connectors for Self-Reconfiguring Robots", Proceedings of International Conference on Robotics and Automation, Washington DC, USA, 2002.

[10] M. Rubenstein, K. Payne, P. Will, W.-M. Shen, "Docking Among Independent and Autonomous CONRO Self-Reconfigurable Robots", Proceedings of International Conference on Robotics and Automation, New Orleans, USA, 2004.

[11] M. Silverman, D.M. Nies, B. Jung, G.S. Sukhatme, "Staying Alive: A Docking Station for Autonomous Robot Recharging," Proceedings of *International Conference on Robotics and Automation*, Washington DC, USA, 2002.

[12] B. Khoshnevis, P. Will, W.-M. Shen, "Highly Compliant and Self-Tightening Docking Modules for Precise and Fast Connection of Self-Reconfigurable Robots," Proceedings of International Conference on Robotics and Automation, Taiwan, 2003.

[13] K. Payne, B. Salemi, W.-M. Shen, and P. Will, "Sensor-Based Distributed Control for Chain-Typed Self-Reconfiguration," Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, 2004.

[14] K. Roufas, Y. Zhang, D. Duff, M. Yim, "Six Degree of Freedom Sensing for Docking Using IR LED Emitters and Receivers," Experimental Robotics VII, Lecture Notes in Control and Information Sciences 271, D. Rus and S. Singh Eds. Springer, 2001.

[15] A.K. Jain, M.N. Murty, P.J. Flynn, "Data Clustering: A Review", ACM Computing Surveys, 31(4), pps. 264-323, 1999.

[16] D.M. Lane, HyperStat Online Tools Website, http://davidmlane.com/hyperstat/normal_distribution.html, Oct. 2, 1999.