A Scalable and Distributed Model for Self-Organization and Self-Healing

(short paper)

Michael Rubenstein
University of Southern California
Information Sciences Institute
mrubenst@usc.edu

Wei-Min Shen
University of Southern California
Information Sciences Institute
shen@isi.edu

ABSTRACT

As the ability to produce a large number of small, simple robotic agents improves, it becomes essential to control the behavior of these agents in such a way that the sum of their actions gives rise to the desired overall result. These agents are modeled as homogeneous, distributed robots, with only one simple short range sensor. Our simple agents are tasked to form and hold a desired swarm shape, independent of the total number of agents. If this shape is damaged by the removal of some of the agents, the remaining agents will recover the former shape, but on a smaller scale. These shapes can also have a pattern such as a picture or drawing displayed on them by controlling the individual robots color, symbolically representing the differentiation of agents within the swarm. This pattern will resize to fit the existing swarm. With the ability to synchronize in time, the swarm gains the ability to change the pattern displayed, resulting in a moving image.

Categories and Subject Descriptors

I.2.11 **[Distributed Artificial Intelligence]**: multi agent systems
I.2.9 **[Robotics]**

General Terms

Algorithms, Design, Reliability, Theory.

Keywords

Robots, swarms, self-healing, self-organization.

1. INTRODUCTION

As the ability to produce large numbers of simple robots improves, it becomes increasingly important to control the robots behavior. One application of a large group of robots, often called a swarm, is to form and hold a global shape that is composed of these small, simple robots, analogous to how cells form the body of an animal. This shape may empower the group of robotic agents to complete a task as a whole, such as in reconfigurable robots [1] and sensor networks [2], that would be difficult or impossible for them to accomplish as a single agent. This swarm allows some interesting additions in functionality, which are not easily possible through a single robot, such as the ability to self-heal. In the event of damage, or removal of some robots from the swarm, the proficiency of the group to complete the desired task may be reduced, or be completely removed. It is easy to imagine damage in a sensor network that could form isolated subgraphs, disconnecting some sensor nodes from transmitter nodes, thus generating incomplete information. Similarly, damage in a reconfigurable robot could remove an essential part such as an arm or leg, preventing it from moving in a desired manner. If these swarms can recover their shape after the damage occurs, then it may also recover its swarm functionality, making self-healing a quintessential part of swarm applications.

Examples of self-healing are ubiquitous in nature, and come in different styles. With healing that uses cell reproduction, called epimorphosis, a salamander, a starfish, or a lizard, can re-grow a lost limb or tail, but their body remains unchanged. The re-grown limb will be the same scale as the original one. In contrast, when a small invertebrate called the hydra shown in figure 1a is bisected, the separated chunks will reorganize through cellular movement (morphallaxis). The existing cells then become complete hydras with the original shape and structure, but in a scale proportional to the number cells that are available (Figure 1b). We define this as *scalable self-healing*. When a structure is dissected, each piece can self-heal into the original structure/pattern but into a smaller scale with the same shape and density of agents. In addition, this type of self-healing mechanism does not require growth or addition of new cells.

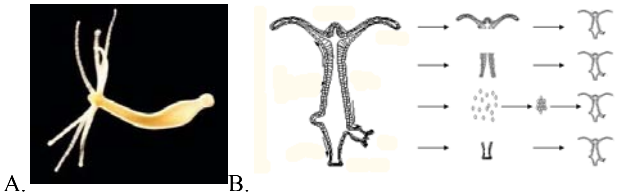
A. B.

Figure 1. (A) A hydra[12] (B) An example of hydra self healing, showing how a whole can self-heal from a piece or a collection of tissues as small as 1/100th the original body size[13]

Currently most artificial self-healing systems envision using a large swarms of simple, small, distributed, homogeneous robots similar to the robots found in [1][3][4] to form the desired self-healing structure, analogous to how biological cells make up an organism. The challenge faced by most of these systems is to first develop a method for identifying the location of a robot within the swarm or structure, and second, how to determine actions for each robot in the swarm that contributes to forming or healing the desired shape. This process is further complicated by the limitations of the robots. The robots are envisioned to be simple in order to keep cost/complexity down, enabling their production in large numbers. The robots also tend to be homogeneous, again to keep cost and complexity down. To allow swarm recovery from damage to any robot, the algorithms tend to be distributed, preventing the loss of a coordinating leader robot due to damage.

The method for accomplishing self-organization and self-healing developed in [5] is impressive, in that it can grow and heal any

Cite as: A Scalable and Distributed Model for Self-organization and Self-healing (Short Paper),Michael Rubenstein, Wei-Min Shen, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., 2008, Estoril, Portugal, pp. 1179-1182.

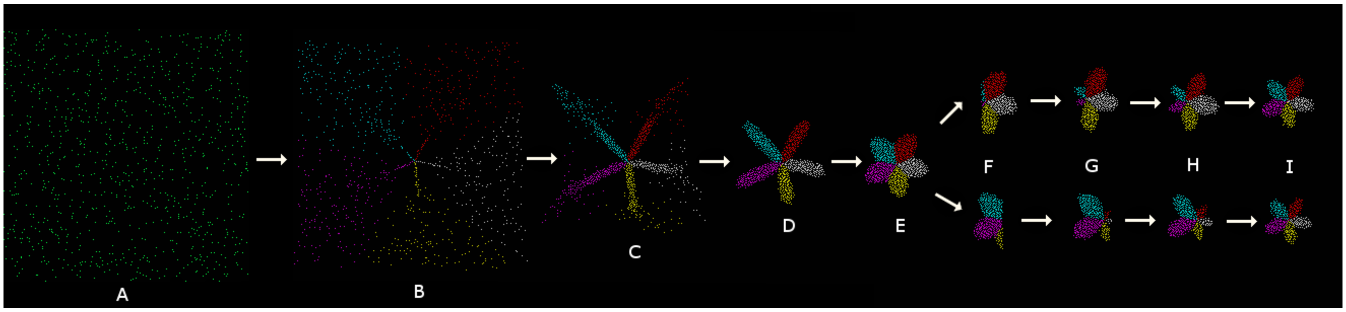

Figure 2. self healing starfish shape formation and recovery after damage and separation.

shape. However, it requires the robots to reproduce, and the field of self-replication in artificial systems such as robots is still in its infancy at best [6][7].

Stoy et al. [8] developed a method for self-healing that is capable of forming and healing an impressive number of 3D shapes; it makes use of direct messaging from one agent to another, i.e. a private communication path between connected robots, increasing the complexity of the robots. They also require a starting seed to start formation, a form of centralization.

The work in [9], by Nagpal et al., simulates very simple robots capable of forming and holding 2d shapes; however, they use some global sensors. For example, to detect the orientation of the robots, a "compass" is used. They also cannot adjust the size of the shape formed to optimally use the remaining robots after damage; they keep the shape the same size, just less dense, i.e. not scalable self-healing.

In this paper, we propose a method for self-healing which is geared towards emulating morphallaxis on a swarm of very simple, physically realizable, homogeneous, distributed agents. They use only one sensor, and this sensor is also used for communication. These agents can form a swarm without any centralized functions, such as a seed, that produces a class of shapes. This shape is correctly sized for the number of agents available by adjusting its scale. They can also demonstrate the ability to differentiate based on location in the swarm, and time. This differentiation is shown by the robot displaying a color. In this paper, we will discuss the model and assumptions used for our simulated robots, as well as show some example results from simulated experiments. We will then go in to detail about the methods used to produce self-healing and pattern formation in the swarm of simulated agents seen in the results, and then conclude.

2. MODEL

Our methods for self-healing and self-organization were tested on a swarm of simulated "puck" robots with finite computational capability, existing in a simple 2D planar world. The assumptions made in the model are chosen to closely emulate the properties of living cells, and are constrained to be as similar to the capabilities of modern robotics/electronics as possible.

These simulated homogeneous robots are identical in every way possible, and even lack a unique identifier. Each robot has one simple exteroceptive (responds to external stimuli) sensor, capable of directly communicating with nearby neighboring robots, and measuring the distance of that communication path. While this sensor is currently simulated as noise-free, previous work [9,10] shows that noisy sensors will not greatly affect results. This simple sensor is the only sensor available to the robots. This means that the robots have no global knowledge of their environment, such as a compass or GPS. Each robot has two degrees of freedom for moving along the plane, and can move up to a maximum speed. In order to display a pattern, each robot can dynamically change its color to one of its choice.

The robots initially are placed at random positions and orientations with respect to world and each other. The robots have no initial knowledge about their location or orientation in this world. They are modeled as a finite sized circle that won't overlap other robots. Each robot has a clock that runs at the same rate as the other robots, however, these clocks are initially started at random times, i.e. not synchronized. The simulation view port is fixed above the 2D plane, looking down. Each robot is represented as a pixel of its chosen color. The simulation runs in discrete time steps. During each time step, T, each robot will communicate with its neighbors, during which it will receive information from their T-1 time step. The position of each robot is then updated on the 2D plane, along with their displayed color.

3. RESULTS

The rules described in this paper allow a swarm of robots with the properties described above to exhibit scalable self-healing and pattern formation. These rules allow a swarm whose robots are placed in random starting locations (figure 2A) to move to form the desired shape, in this example, a five armed "star fish" (figure 2B-E). Between figure 2E and 2F, the swarm is cut in half and the halves separated beyond communication range. The robots continue to move in accordance of the self healing rules, and in figures 2F-I, the separated swarms independently re-form the desired shape, but in a smaller scale, resulting in two smaller but proportionate "star fish".

Through the use of each robot's individually displayed color, the swarm is capable of differentiating by displaying a pattern upon the shape which is being maintained. This pattern can be a simple coloring scheme; for example, the star fish in figure 2, has a unique color for each arm. In figure 3, there is a more complicated coloring scheme, where the swarm displays a picture of the earth. The pattern can be any arbitrary picture. As shown in both cases, if the swarm is damaged, as the robots move to reform the damaged shape, they will also change their individual colors in order to re-form the desired pattern. The new pattern will automatically re-size to fit the original pattern upon the new smaller swarm, as shown in figure 2 and 3.

With the addition of a method for synchronizing the robots in time, it becomes possible not only to display a pattern on the robots, but also a time varying pattern. This could be in the form of a movie, or as text that scrolls across the swarm of robots. Figure 4A shows an example of the "scrolling text" displayed on a swarm holding a circle shape. As with the self healing pattern, this time varying pattern will also recover from damage, and scale to the size of the swarm, as shown in figure 4B.

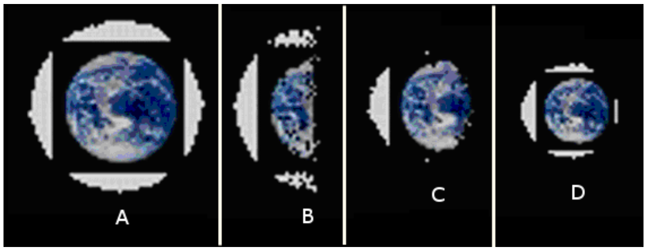

Figure 3. (A) Pattern self healing in a circle shaped swarm, (B) immediately after damage, (C-D) recovering from damage. Note: robots with no color appear to be same color as background

Figure 4. (A) Images of a circular swarm during increasing time, showing the text scrolling across the swarm. (B) Scrolling text swarms before damage (left), after damage once (center), and after damage twice (right).

4. METHODS

The self healing/organization behavior in each robotic agent is separated into two concurrently running tasks. The first task is to setup and maintain a coordinate system that uniquely and correctly identifies the location of each robot. Second, each robot makes decisions and movements to form or reform a predefined shape (section 4.2), pattern (section 4.3), or time-varying pattern (section 4.4), making use of the coordinates determined in the first task. The task of setting up and maintaining a coordinate system is done using the available range sensor, and communication between neighboring robots through a process called trilateration, detailed in section 4.1. The method used to produce a coordinate system is independent to the rest of the self-healing, and could actually be produced using any method desired; however, trilateration is capable of forming a coordinate system using the simple sensor we envision on the robots. Other methods may require more complicated sensors.

4.1 Trilateration

The robots are capable of determining their relative positions to each other using a form of trilateration. This iterative process starts with each agent forming a random initial guess as to its location in a coordinate system (x_{self},y_{self},z_{self}). In every iteration, through communication, the agent will receive its neighbors estimated values for their own (x_j,y_j,z_j), for neighbors within sensor range. The distance for that communication ($d_{j,self}$) is also measured with the sensor. These values for each neighbor will be used in the trilateration formula 1, which is minimized using a gradient method. This minimization results in values for (x_{self},y_{self},z_{self}) that reduce the error of the guessed distance between (x_{self},y_{self},z_{self}) and (x_j,y_j,z_j) when compared to the actual measured distance, $d_{j,self}$. A similar method of trilateration has been used before for localization of robot groups [9,10], and in a statistical method multi-dimensional scaling [11]. Formula 1 differs from other trilateration formulas in that it allows three degrees of freedom even though the robots exist on a plane. The formula contains a forcing term $|z_{self}|$ that pushes the three dimensional solution onto a XY plane. It has been observed experimentally that this modification allows the swarm to converge on a correct coordinate system quicker and with less communication than without it.

$$\sum_{j=1}^{All\ Neighbors}\left(d_{j,self}-\sqrt{\left(\left(x_{self}-x_j\right)^2+\left(y_{self}-y_j\right)^2+\left(z_{self}-z_j\right)^2\right)}\right)^2+|z_{self}|$$

Formula 1. Modified trilateration formula.

There is no constraint to this coordinate system that is agreed upon through trilateration, due to the complete lack of global knowledge. Because of this, the origin and axis of the coordinate system could be in any orientation and is equally likely to be left handed or right handed when viewed from the global viewport. To demonstrate this, the top of figure 5 shows a desired shape/pattern, while the squares below it show images taken from multiple simulation runs, with various orientations and handedness. An agent can determine its orientation with respect to the agreed-upon coordinate frame by making two movements, orthogonal to each other, and finding its new coordinates after each movement. After its orientation in the agreed upon coordinate system is known, a robot can move in a desired direction with respect to that coordinate system.

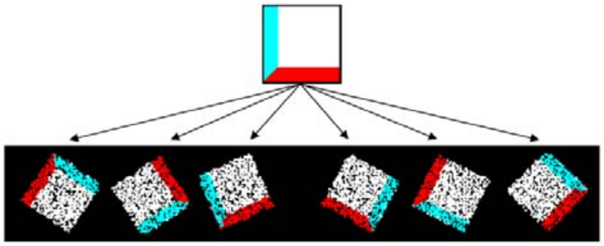

Figure 5. Demonstrating the arbitrary orientation or the agreed-upon coordinate system discovered through trilateration.

4.2 Shape formation

Currently, the target shape to be formed by the swarm, $f_{shape}(\theta)$, is constrained to any shape that can be described as a smooth positive polar function from 0 to 2π. After determining its position (r_{self}, θ_{self}) in the agreed upon coordinate system, an agent will calculate the negative direction of the surface normal of $f_{shape}(\theta_{self})$. It will try to move in that direction, and sense if a collision occurs. If a collision is detected (by sensing no change in its (x_{self},y_{self},z_{self}) values from trilateration after commanding a movement), the robot will just move in a random direction, and then repeat the process. This movement rule will cause the swarm to form the desired shape in the scale appropriate to the number of agents. This movement can be thought of as the robots existing in a potential field in the form of the desired shape. The robots will try to move inward in the direction of the field gradient, towards the lower potential. In figure 6, a visualization of the potential field for some sample shapes can be seen, where the darker colors represent lower potential, or a lower multiple of the shape function. Following this rule, the robots will ensure that all the available positions with a potential lower than its current potential will be filled by another robot.

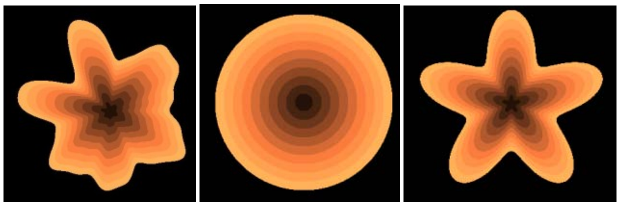

Figure 6. potential field for some sample shapes

4.3 Pattern formation/self-healing

To display a pattern over the shape of the swarm, like shown in figure 3, each agent chooses an appropriate color. Agents determine the scale of the shape, which is the largest scalar multiple ($R_{\max}$) of the shape function to enclose every agent in the swarm. This is calculated iteratively as follows,

$$R_{\max} = \max\left\{ \sum_{j=1}^{N} \left(\frac{R^{j}_{\max}}{N} \right), \left(\frac{r_{self}}{f_{shape}(\theta_{self})} \right) \right\}_{\text{where } N \text{ agents are in range}}$$

Where each agent updates its $R_{\max}$ value to either the average $R_{\max}$ of its neighbors or to its own scale (r_{self} / $f_{\text{shape}}(\theta_{\text{self}})$), whichever is larger. A small random percentage of the time, an agent will reset its $R_{\max}$ value to be its own scale, in order to allow the scale of the shape to shrink in the event of damage. Each agent will then use its current $R_{\max}$ to determine its color to display using the function

$$F_{\text{pattern}}(r_{\text{self}}, \theta_{\text{self}}, R_{\max}) \rightarrow \text{color}$$

This pre-defined function F_{pattern} tells the robot what color it should display for its given position (r_{self}, θ_{self}), and the detected size of the swarm ($R_{\max}$).

4.4 Time-varying pattern

For demonstration of behaviors that are synchronized in time, all agents are given a rule to automatically synchronize its internal clock with those of its neighboring agents. The formula for this rule is:

$$t_{self} = \sum_{j=1}^{N} \left(\frac{t_j}{N} \right)_{\text{where } N \text{ agents are in range}}$$

With this iterative rule, every agent's clock, t_{self}, becomes synchronized to that of its neighbors by setting its clock to be the average of its neighbor's clocks. The function for determining its color is modified as:

$$F_{\text{pattern}}(r_{\text{self}}, \theta_{\text{self}}, R_{\max}, t_{\text{self}}) \rightarrow \text{color}$$

This modified function F_{pattern} tells the agent what color it should display for its given position (r_{self}, θ_{self}) time (t_{self}) and the current scale of the swarm ($R_{\max}$). Time changing color patterns, such as the scrolling text shown in figure 4, and movies, can be found at http://www.isi.edu/robots/self_heal_html/index.htm

5. CONCLUSION AND FUTURE WORK

The model described in this paper is motivated to have minimal artificial requirements and to be consistent with the conditions found in nature. It shows that even with simple sensors, and no global information, it is possible to control, organize, and differentiate a large swarm of simple agents. It may help us to better understand how natural organisms self-form and self-heal, and provide new insights for developing large scale artificial distributed organizations composed of simple robotic and/or software agents. Our future work includes the generalization of shape functions to cover a larger range of spatial patterns. We would also like to modify the existing model assumptions to match that of existing robotic hardware, or hardware of our design. Modification of the model assumptions could also be geared towards better emulating biological self-healing and pattern formation, in order to better understand natural phenomena.

6. REFERENCES

[1] M. Yim, W. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, G. Chirikjian. Modular Self-Reconfigurable Robot Systems -- Challenges and Opportunities for the Future. IEEE Robotics and Autonomation Magazine, March():43–53, 2007.

[2] A. Mainwaring, R. Szewczyk, D. Culler, J. Anderson Wireless Sensor Networks for Habitat Monitoring. ACM International Workshop on Wireless Sensor Networks and Applications (WSNA), 2002.

[3] Tolley, M., Zykov, V., Lipson, H., Erickson, D., (2006) "Directed Fluidic Self-Assembly of Microscale Tiles" Micro-Total Analysis Systems (uTAS) 2006, Tokyo, Japan, Oct. 2006.

[4] J. Bishop, S. Burden, E. Klavins, R. Kreisberg, W. Malone, N. Napp, and T. Nguyen. Self-organizing programmable parts. In *International Conference on Intelligent Robots and Systems*. IEEE/RSJ Robotics and Automation Society, 2005.

[5] A. Kondacs, Biologically-inspired Self-Assembly of 2D Shapes, Using Global-to-local Compilation, International Joint Conference on Artificial Intelligence (IJCAI), 2003.

[6] Suthakorn, J., Andrew B. Cushing, and Chirikjian, G.S., An Autonomous Self-Replicating Robotic System Proceedings of the 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2003

[7] Lipson H. (2005) "Self-reproducing machines", Nature Vol. 435 No. 7038, pp. 163-164

[8] K. Stoy, R. Nagpal. Self-Repair Through Scale Independent Self-Reconfiguration. International Conference on Robots and Systems, (IROS), Sendai, Japan. 2004.

[9] J. Cheng, W. Cheng,R. Nagpal, Robust and Self-repairing Formation Control for Swarms of Mobile Agents, National Conference on Artificial Intelligence (AAAI '05), July 2005.

[10] R. Grabowski, P. Khosla. Localization Techniques for a Team of Small Robots. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'01).

[11] I. Borg, Modern Multidimensional Scaling: Theory and Applications. Springer Series in Statistics. 2005.

[12] Vince G., Explosive sting of jellyfish captured on film. NewScientist may 9 2006

[13] Shimizu H., Sawada Y., Sugiyama T., *Dev Biol.* **155**, 287 (1993)