

# Simulation of Artificial Ant's Behavior in a Digital Environment

Paulo E. Merloti, Joseph Lewis

**Abstract**—Ants present a very good natural metaphor to evolutionary computation. While each individual's computational power is small compared to more evolved species, it is the power of their colonies that inspire computer scientists. This paper presents "Ant Box Simulator", a software that allows the simulation of ants in a restricted environment. The simulator demonstrates how digital ants make use of an autocatalytic behavior in order to collectively find a solution for a posed problem in reasonable time. We also show that while pheromones play an important role on biological ant's navigation, it is not the only sense they rely on.

**Index Terms**—Swarm intelligence, artificial life, ant algorithms, emergence.

## 1 INTRODUCTION

Ants are relatively simple beings. With their small size and small number of neurons, they are incapable of dealing with complex tasks individually. The ant colony on the other hand is often seen as an "intelligent entity" for its great level of self-organization and the complexity of tasks it performs. In this paper, we will focus on one of the resources ant colonies use for their achievements, pheromone trails. We will try to show some relationship between the stigmergic behavior facilitated by pheromones and the process of representation in a complex system.

One way ants communicate is by secreting chemical agents that will be recognized by receptors on the bodies of other ants. For example, an ant is capable of determining if another ant is a member of its own colony by the "smell" of its body. One of the most important of such chemical agents is the pheromone. Pheromones are molecules released from glands on the ant's body. Once deposited on the ground they start to evaporate, releasing molecules of that chemical agent into the air. Ants inside the pheromone's cone of evaporation [1] are attracted by that pheromone. Foraging ants leave a trail of such scent, which stimulates other ants to follow that trail, dropping pheromones while doing so. This use of the environment as a medium for indirect communication is called *stigmergy*. This process will continue until a trail from the ant colony to the food source is established. The creation of a trail with the shortest distance from nest to food source is a side-effect of their behavior—not something they have as

an *a priori* goal. While following very basic instincts, ants accomplish complex tasks for their colonies in a perfect demonstration of emergent behavior. In the foraging example, one of the characteristics of the pheromone trail is that it is highly optimized, tending toward the shortest highway between the food source and the ants' nest.

Such behavior was once studied in laboratories in order to find out how pheromones were used by ants. In a very inspiring experiment, Deneubourg et al. [3] demonstrated how ants use pheromones to optimize the roundtrip time from nest to food source. In this experiment denoted the "double bridge", ants in the nest are separated from the food location by a restricted path with various branches of different lengths. As ants travel the trail back and forth they leave their scent trail behind. The experiment shows that pheromone concentration is higher on the shorter path, therefore increasing even more the probability that an ant will choose the shortest path.

It is, however, naïve to think that pheromones are the only resource ants use to navigate. In another experiment, Bethe [4] proves that pheromone trails are not polarized, as was once thought to be the case. The experiment consists of a nest, a food source and a pivot table between food location and the ant's nest. After a pheromone trail is formed over the pivot table, one ant is released from the nest. While the ant is in the middle of the trail, the pivot table is turned 180°. If the ant would keep its heading, it would end up in the nest again as the table was rotated, but amazingly, the ant also turns its direction and ends up in its original destination, the food source. This experiment demonstrates that ants also depends on other senses to navigate, such as position of the sun in the sky (or a strong enough light source), gravity, slope etc..

- P.E. Merloti (presenter) is with the San Diego State University, San Diego, CA 92182. Postal address: 2295 Cabo Bahia, Chula Vista, CA 91914. E-mail: [padu@ieee.org](mailto:padu@ieee.org). Telephone: +1 619 656-0048, Fax: +1 858 550-7330
- J. Lewis is with the San Diego State University, San Diego, CA 92182. Postal address 5500 Campanile Dr. MC7720, San Diego, CA 92182. E-mail: [lewis@cs.sdsu.edu](mailto:lewis@cs.sdsu.edu). Telephone: +1 619 594-2014 Fax: +1 619 594-6746

## 1.1 Existing work

The Artificial Intelligence community is seeing a shift toward techniques based on evolutionary computation. Inspiration comes from several natural fields such as genetics, metallurgy (simulated annealing) and the mammal immune system. Growing interest in ant colony and swarm algorithms is further demonstration of this new trend.

Marco Dorigo leads the research on optimization techniques using artificial ant colonies [5]. Since 1998, Dorigo has been organizing a biannual workshop on Ant Colony Optimization and swarm algorithms at the *Université Libre de Bruxelles*. Dorigo and his colleagues have successfully applied ant algorithms to the solution of difficult combinatorial problems, such as the traveling salesperson problem, the job scheduling problem and others. Ramos [6] and Semet [7] use the ant colony approach to perform image segmentation. Heusse et al. [8] applies concepts of ant colonies on routing of network packages. A more detailed summary of these studies can be found in a summary paper [9].

In simulation, ant colony behavior offers clear demonstration of the notion of emergence—that complex, coordinated behavior can arise from the local interactions of many relatively simple agents. Stigmergy appears to the viewer almost intentional, as if it were a representation of aspects of a situation. Yet, the individuals creating this phenomenon have no awareness of the larger process in which they participate. This is typical of self-organizing properties: visible at one level of the system and not at another. Considering this, Lawson and Lewis [10] have suggested that representation emerges from the behavioral coupling of emergent processes with their environments. We hope here to reveal, through experiments with a simple ant colony, the variety of parameters which affect this self-organizing tendency.

## 1.2 Introduction to Ant Box Simulator

The *Ant Box Simulator* (ABS) idea started from curiosity about ant colony behavior and the amazing feats they demonstrate. The concept of the simulator is of a two dimensional digital box where ants are represented by small objects bounded by a perimeter. They are inserted in the environment around a spot denoted “IN Hole” and their goal is to find the “OUT Hole”. Ants have a limited sensorial radius to “smell” pheromones or to detect the OUT Hole. The probability of one ant finding the exit of the box is directly related to area of the arena and the distance from IN and OUT holes. The next sections of this paper demonstrate a series of experiments using the simulator by varying several parameters and strategies that affect ant’s behavior.

## 2 ARE PHEROMONES GOOD ENOUGH?

As stated previously, ants rely heavily on pheromones to guide themselves. Most ant colony algorithms utilize the pheromone concept and a highly parallel architecture to solve hard problems. It is true that being computer scientists seeking *inspiration* to solve engineering problems, we don’t need to be fixated on a high fidelity model of an ant colony. However, it is very important to keep in mind that our inspiration comes from a reduced biological model of how ants navigate.

The fundamental question we try to answer here is: “Can we solve the problem or searching for the OUT hole using only the pheromone concept?” Are pheromone trails a good enough metaphor to solve the proposed problem? Moreover, with these experiments we can ask what other parameters in this computational model provide the flexibility to produce useful behavior in the context of various problems.

As a baseline, we know that in other lines of research, the pheromone concept has proven useful, achieving excellent results in combinatorial optimization problems for example [5].

## 3 SIMULATOR ARCHITECTURE

The ABS is an application that runs on Windows 32 platform. Due to its demand for graphic computation, a Pentium IV or greater with a fast video card is recommended. Also, it relies on platform specific DirectX technology, version 7 or higher. The software was developed using Borland’s Delphi, a Pascal based language. The main reason for this choice was high productivity and fast performance of native code offered by this tool.

The application was built with expansion in mind; therefore the object model (fig. 1) implements not only the Ant concept, but a digital environment where objects and even other insects could be placed together in later experiments. Following is a brief explanation of this object model.

The Main Form is responsible for drawing, creating and managing all kinds of simulation objects, as well as presenting a user interface for interaction with parameters. It keeps a dynamic list of simulation objects on SimsList. All simulation objects are descendants of TSimObject. TSimObject is responsible

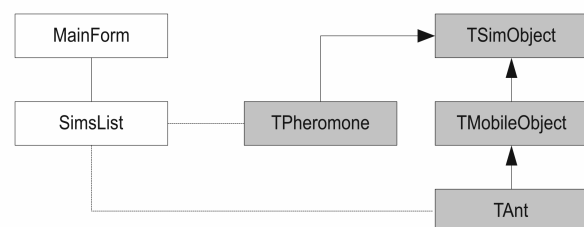


Fig. 1. Ant Box Simulator Object Model.

TABLE 1  
ANT BOX SIMULATOR MAIN PARAMETERS

Symbol	Parameter	Experiment Value
$\alpha$	Ant direction max variance	80
none	Pheromone duration	30
none	Pheromone reinforcement	40
none	Sensorial radius	45
$\pi$	Searching radius	90
$\beta$	Walk before pause	45
none	Pheromone spacing	35
none	Ant follows pheromone	YES
none	Ant follows reinforced pheromone	YES
$\gamma$	Searcher follows reinforced trail	YES
$\delta$	Drop pheromone while following trail	YES/NO
$\sigma$	Stochastic pheromone following	YES
none	Change direction only when stopped	NO

for basic aspects of the object, such as position, size, identification and a virtual method for drawing. A TMobileObject implements basic animation methods such as wall collision, collision with other objects, current speed and direction. An ant (TAnt) is a specialization of a TMobileObject that refines and implements navigation techniques using stochastic patterns and pheromone-following algorithms. Ants may release pheromones (TPheromone) into the environment. Pheromones have the property that they expire after a certain amount of time; their strength is proportional to their remaining life time.

The digital environment for the experiments is a two dimensional array of pixels representing the surface inside the box with an approximate area of 644,496 squared pixels. The IN hole is located at the center of the box, and the OUT hole is located at the right bottom corner distant 535 pixels.

There are two types of ants in the ABS, regular ants and searching ants. When regular ants find the

OUT hole, they exit no matter what. Searcher ants will always leave a pheromone trail, and if the OUT hole is found, then they will trace back their trail reinforcing their own pheromones. If they find a wall instead, they will destroy their trail. The need for different kinds of ants (or different kinds of pheromones) has been seen elsewhere [10].

## 4 EXPERIMENT DESIGN

Experiments are run for 300 seconds, and performance is measured by comparing the number of ants that have not yet exited the box during the time allotted for the experiment. All the experiments start ticking the clock when the first batch of 100 regular ants is released through the IN hole.

Our first experiment (*xp1*) is a random search; an initial batch of 100 ants is released into the environment to run their random walk, and some of the ants will find the OUT hole by chance. *xp1*, *xp2* and *xp3* were performed 3 times each. In *xp2*, batches of 10 searching ants are released 10 seconds apart, until 250 seconds of box time. Experiment 3 (*xp3*) is very similar to *xp2*, except that it releases only one searching ant every 10 seconds until the 250 seconds mark is reached. Experiment 4 (*xp4*) and 5 (*xp5*) are based on *xp2* and *xp3* respectively, with the difference that regular ants will also drop pheromones when they find a pheromone trail. For reasons to be seen later, these last two experiments were run only one time each.

### 4.1 Parameter Definition

The simulator offers several parameters that affect the individual behavior of an ant. Table 1 describes the initial value of such parameters. In these experiments, we will investigate the consequences of varying one of these parameters, namely  $\delta$ . Additionally we will also investigate how different searching strategies on scheduling batches of searching ants affect the result of the experiments. The other parameters directly

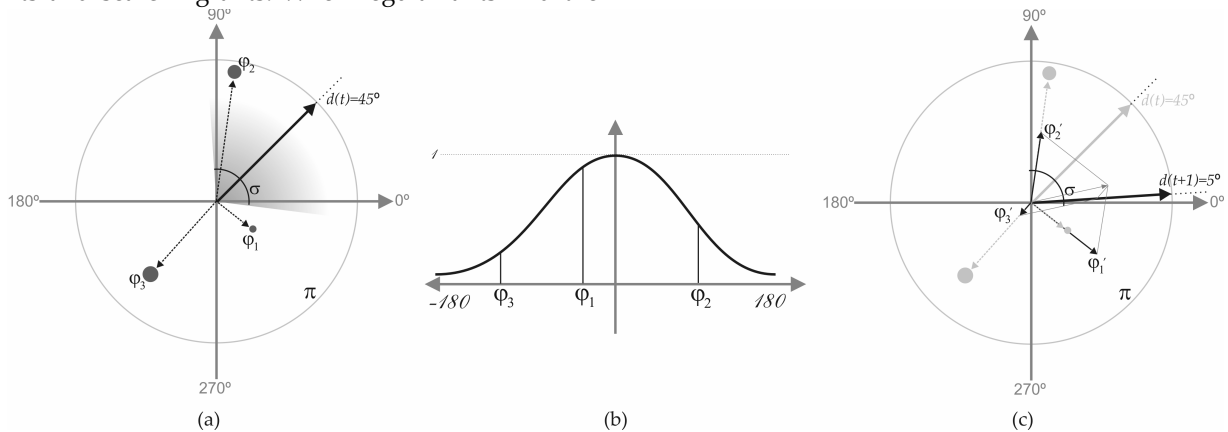


Fig. 2. Geometric representation of how the simulator calculates a new direction for one ant when pheromones are within sensorial radius. (a) represents 3 pheromones within the ant's sensorial radius. (b) gaussian adjustment of pheromone weight based on strength, pheromone distance from ant and angle between  $\phi_1$  and current direction  $d(t)$ . (c) the new direction is the sum of the three weighted pheromone vectors.

affect how ants navigate through the environment, and they are better explained in the next section. All together we hope to gain insight into which parameters to vary, what are appropriate values, and what are the consequences of those settings for the self-organizing behaviors they induce.

## 4.2 The Mathematics of Navigation

Before we proceed to the experiments, let us investigate how navigation and pheromone following is implemented in the ABS. The ant's reference system is self-centered and the direction they are headed to is measured in degrees, with 0/360 degrees mark pointing to the east side of the screen and 90 degrees pointing north.

When ants are released into the box, they immediately start their random walk inspired by "Brownian movement" [11]. Ants follow a straight line of random length (limited by  $\beta$ ) headed to some initial random direction. When they reach the end of this segment, a new direction ( $d(t+1)$ ) is chosen according to (1). Note that in (2) we assume that function "Random( $n$ )" returns an uniform random number in the range  $[0..n]$ .

$$d(t+1) = d(t) + \theta \quad (1)$$

Where

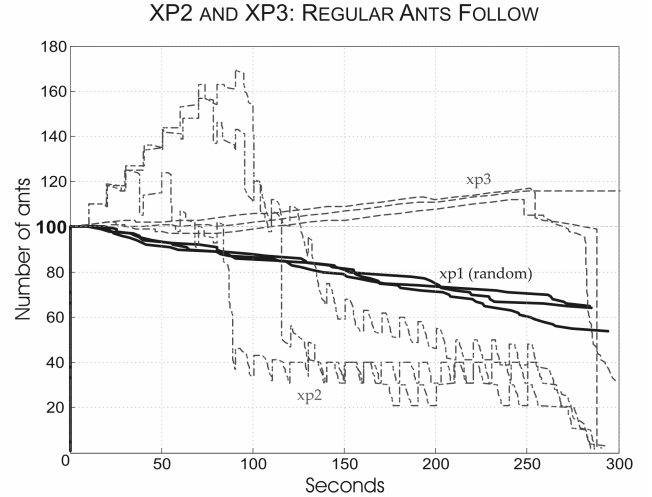
$$\theta = \text{Random}(\alpha) * \text{Random}(2) - 1 \quad (2)$$

When one or more pheromones are within the ant's sensorial radius, the new direction is calculated using a different method. When  $\sigma$  is enabled a stochastic pheromone following technique is employed (see fig. 2); otherwise the new direction is a straight line from the ant's current position to the strongest pheromone within the sensorial radius.

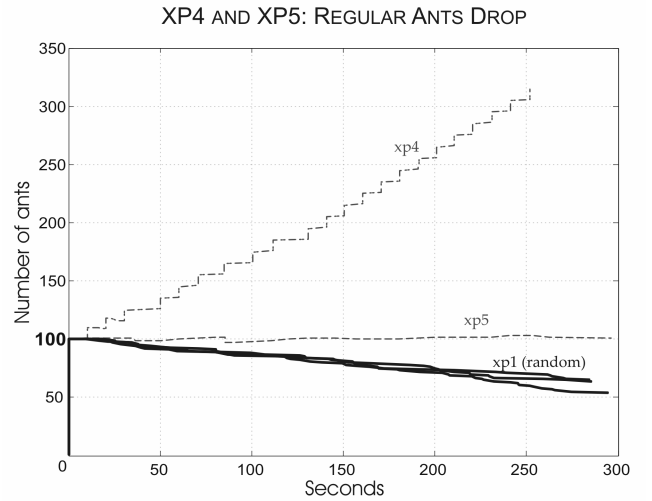
From the ant's point of view, there are three important attributes related to a pheromone: estimated direction, distance and strength. Biological ants don't have perfect sensors; additionally natural factors may contribute to erratic pheromone sensing (i.e. wind). Therefore a mechanism to introduce noise is necessary. A second feature modeled is that ants have a better sense for pheromones located in front of them (because of the location of their antennas). Therefore pheromones located on the relative north of the ant have a greater weight on their choice for a new direction.

Fig. 2 depicts a situation where a given ant detects three pheromones within its sensorial radius. Each pheromone vector is attenuated or intensified based on its strength (represented by the diameter of the dots in fig. 2), the distance between pheromone location and ant center, and the angle between the ant's current direction and pheromone direction. The new calculated direction is the sum of all of the adjusted pheromone vectors.

Also, in order to simulate the uncertainty factor found in nature, we must introduce some random noise into the system. Therefore, the new direction



(a)



(b)

Fig. 3. In both graphs, xp1 represents the random search, where 100 ants are let loose to find the exit hole at random. No ant is released after  $t=250$ . (a) Regular ants do not drop pheromones while following a trail. In xp2, batches of 10 searching ants are released every 10 seconds. In xp3 only one searching ant is released every 10 seconds. (b) Experiments xp4 and xp5 reproduce xp2 and xp4 respectively with the only exception that regular ants drop pheromones while following a scent trail.

can be calculated as follow:

$$d_{pher.}(t+1) = \sum_{i=1}^N \mathbf{v}_i w_i + r e^{-\frac{1}{2} \left( \frac{2r}{180} \right)^2} \quad (3)$$

$$w_i = s_i \cdot \left( \frac{|\pi - d_i|}{\pi} \right) \cdot \left( a_i \cdot e^{-\frac{1}{2} \left( \frac{2a_i}{\alpha} \right)^2} \right) \quad (4)$$

$N$  is the number of pheromones within the sensorial radius ( $\pi$ );  $\mathbf{v}_i$  is a unit vector pointing to  $\varphi_i$ ;  $w_i$  is the weight of the pheromone calculated according to (4). The variable  $r$  introduces the uncertainty factor to the equation. Every time a new direction must be calculated,  $r$  is drawn by subtracting a random number in the  $[-180,180]$  interval by the current direction. The Gaussian

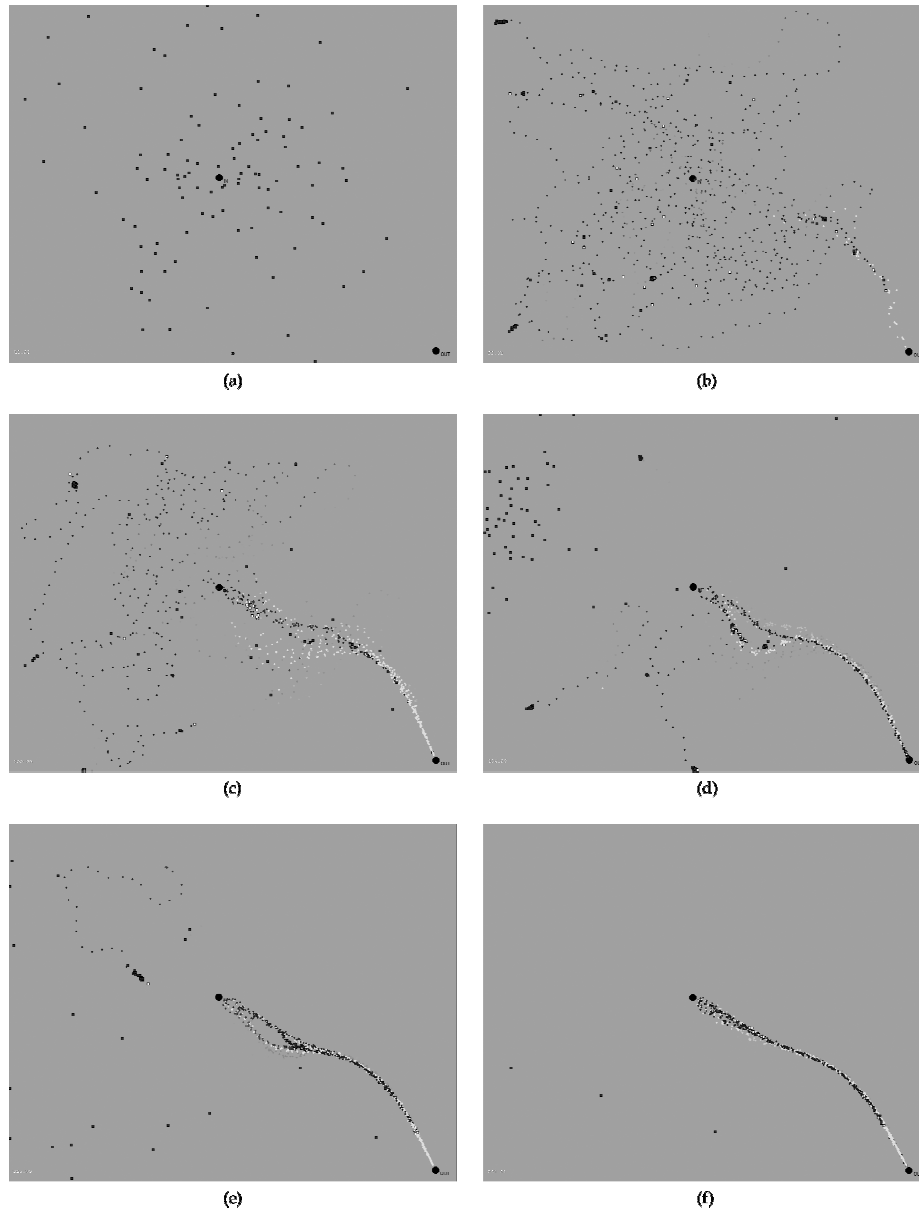


Fig. 4. Screenshot sequence of a run of experiment 2 (xp2). (a)  $t=11s$ : 100 regular ants randomly walking around the box. (b)  $t=81s$ : 8 batches of searching ants were already released, some searching ants already found the exit hole and start reinforcing their trail. (c)  $t=122s$ : first trails start drawing attention of other searching ants. (d)  $t=154s$ : trail consolidating, still plenty of regular ants around. (e)  $t=215s$ : less random ants in the box, trail still shows a bifurcation. (f)  $t=291s$ : trail is unique and as long as searching ants are inserted into the box, the trail is kept alive and keeps optimizing itself. After 300 seconds, no random ants are found in the box.

function implements inertia, as the noise is attenuated if it disagrees by too much with the current direction.

In (4),  $s_i$  is the strength of pheromone  $\varphi_i$ ;  $d_i$  is the normalized distance from  $\varphi_i$  to the ant's position and  $a_i$  is the angle between pheromone  $\varphi_i$  and the ant's current direction ( $d(t)$ ).

## 5 EXPERIMENTAL RESULTS

As stated earlier, the ants in these experiments have a “goal” of finding the exit of the box in the least amount of time. The individual ants' behaviors are not *a priori* organized around that “goal”, but their emergent behavior (as a result of their tendency to

exit on discovery of the OUT hole) is consistent with that goal. There is another goal evident in the emergent behavior (again not determined *a priori*): to create a near optimum (straight line) pheromone trail from the IN hole to the OUT hole. This allows ants inserted after the trail was established to have a reference on how to get out of the box as soon as they enter it. It is in this way that the colony self-organizes so that its behavior over time improves—stigmergic learning. We hope to see by varying the parameters by which the ants achieve this self-organization just which features of the colony have which effect on how the ants behavior represents information about the environment.

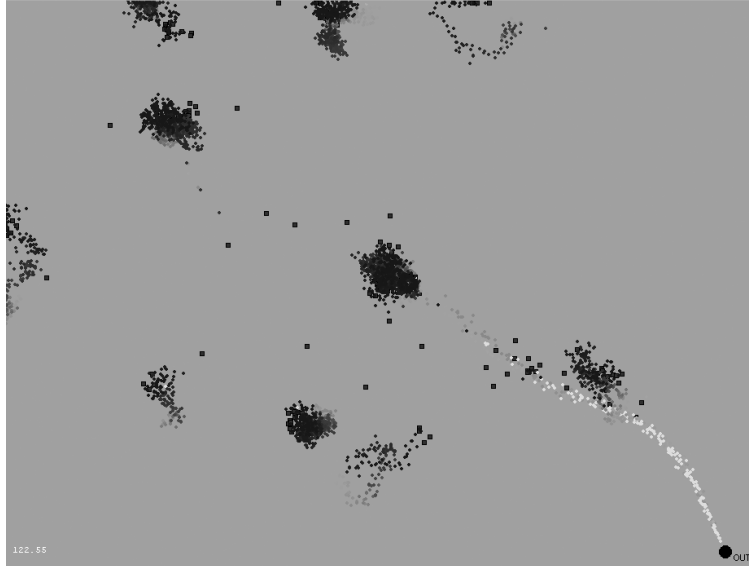


Fig. 5. Even though a few searching ants built a trail from IN to OUT holes, regular ants were never attracted to that trail, since they were too busy trying to follow the local trails that lead to nowhere.

Figure 3 shows a graphic comparing the results of these experiments described in sections 3 and 4. Two aspects must be considered when analyzing the performance of a given experiment. One is how soon the exit is found by a considerable number of ants, which is observed in the graph as a big drop in the number of ants. The second aspect is the dynamics of the number of ants in the arena. In random search (xp1) for example, the total number of ants never went higher than 100 for obvious reasons (no searching ant was ever introduced).

Experiment 2 (xp2) was the best performing one if we seek shortest time to find the exit. As observed, at earlier steps a greater number of ants will be on the box due to the introduction of the extra searching ants, but the exit was found very fast, normally in less than 50 seconds. Another point observed in this experiment is that even after the exit was found, there were a good number of ants still in the box, mainly doing path maintenance. This experiment also demonstrated that as long as searching ants are kept doing trail maintenance, the path from the IN to OUT holes will slowly converge to the quasi-optimum solution (the path varies in small amounts as long as there are ants traversing it).

Experiment 3 (xp3) tells us that if we reduce the input flow of searching ants, it takes much longer to find and establish the trail to the OUT hole, but it is the more economical solution in terms of number of ants in the box. In this case, the solution was found at the fringe of the 5 minutes time limit, and only in two of the three repetitions of the experiment. One should always keep in mind that as the number of searching ants increase, so does the number of pheromones in the box and consequently the computation cost of the experiment.

When we allow non-searching ants to release pheromones when they find a trail (parameter  $\delta$ ), the

situation goes out of control. Ants will tend to form circular groups, where one ant follows the fresh pheromone of the other and vice versa. At this point, pheromone starts to be deposited at very high rates and results are much worse than random search. In random search, ants are free to randomly walk the box. Figure 3 (b) shows the results of xp4 and xp5. Note on xp4 how the number of ants in the box grows indefinitely as ants are captured in this circular pheromone following pattern.

Figure 4 shows a screenshot sequence of xp2 demonstrating the patterns on how a trail is created. Figure 5 shows an example of xp4 and the circular grouping pattern.

The ABS can be found on the webpage at [12], with source code in Delphi available by request.

## 6 CONCLUSIONS

In this paper we introduced the idea of an ABS, a software program that simulates the stigmergic behavior of biological ants when faced with an artificial problem, finding the exit of a two dimensional box.

After running the experiments, we were able to show that pheromones are really useful if used together with a “good” strategy. Also, we were able to see that different strategies may serve different purposes. The settings used in the experiment shown in Figure 4 seems, for that “problem” setting, to be effective. Clearly the settings used in the experiment shown in Figure 5 is not as effective. This and other parameter variations modify the colony behavior notably. Such is the case with almost all demonstrations of emergent computation. It is, however, interesting to keep in mind the notion of exaptation—in which phenomena with no initial merit become utilized by an evolutionary process for

an entirely different purpose. We believe that for a distributed, emergent system like an ant colony (or other emergent systems of wide variety) the variation in parameters is exactly that feature that can be leveraged in an evolutionary wrapping of the emergent system. That is how a system that has no “representation” for some feature of a domain can come to have representation that is effective and novel.

As in other studies about ant algorithms, our artificial ants used only the pheromone concept to guide themselves, and as we saw on xp4 and xp5, this approach can lead to catastrophic behavior sometimes. The main reason why xp4 and xp5 experiments failed is related to the discussion of section 1. Biological ants don't rely only on pheromones to navigate. It would be interesting to research the creation of a framework of ant colony algorithms that include methods inspired on other resources used by biological ants such as gravity, light sources and vision. Also, the enrichment of the simulator environment with obstacles and even other types of insects, possibly predators, is a very interesting idea.

To conclude this paper, we share our feeling that the toughest problem we faced dealing with evolutionary techniques such as ant algorithms is finding the right parameters in order to direct the system to solve a specific problem. If we find a way to pressure the population of such system to change its own parameters and naturally evolve into a body capable of solving a specific problem, then our task would be defining problems in such ways that would be understandable for our population. Perhaps genetic algorithms would be a good approach. Would it then materialize our dreams of a machine capable of solving problems with no necessity of being pre-programmed? Would it eliminate the brittleness problems found on many approaches to artificial intelligence?

These questions are the main focus of research on many AI studies, and in the author's point of view, biological inspired ideas have a great probability of success; after all, for many problems we still don't know how to solve using machines, nature has proven methods that work everyday almost effortlessly.

## REFERENCES

- [1] R.W. Matthews, J.R. Mattheus, *Insect Behavior*. University of Georgie, New York: Wiley-Interscience, 1942.
- [2] D. Gordon, *Ants at Work: How an Insect Society is Organized*. New York: The Free Press, 1999.
- [3] S. Goss, S. Aron, J.L. Deneubourg, J.M. Pasteels, *Self-organized shortcuts in the Argentine ant*. *Naturwissenschaften*, pp. 76:579-581, 1989.
- [4] A. Bethe, *Recognition of nestmates, trails*. *Arch. Gesamt. Physiol.*, pp.70:17-100, 1898.
- [5] M. Dorigo, G. Di Caro, L.M. Gambardella, “Ant Algorithms for Discrete Optimization,” *Artificial Life*, vol.5, no.3, pp.137-172, 1999.

- [6] V. Ramos, F. Almeida, “Artificial Ant Colonies in Digital Image Habitats – A Mass Behavior Effect Study on Pattern Recognition,” *Proc. of ANTS'2000, 2<sup>nd</sup> International Workshop on Ant Algorithms*, pp.113-116, Brussels, Belgium, Sept. 2000.
- [7] Y. Semet, U. O'Reilly, F. Durand, “An Interactive Artificial Ant Approach to Non-Photorealistic Rendering,” *Springer-Verlag, K. Deb et al. (eds.): GECCO 2004, LNCS 3102*, pp.188-200, Berlin, 2004.
- [8] M. Heusse, S. Guérin, D. Snyers, P. Kuntz, “Adaptive Agent-driven Routing and Load Balancing in Communication Networks,” Technical Report RR-98001-IASC, Département Intelligence Artificielle et Sciences Cognitives, ENST Bretagne, 1998.
- [9] P.E. Merloti, “Optimization Algorithms Inspired by Biological Ants and Swarm Behavior,” San Diego State University, Artificial Intelligence Technical Report – CS550, San Diego, 2004.
- [10] J. Lawson and J. Lewis, “Representation Emerges from Coupled Behavior”, *Self-Organization, Emergence, and Representation Workshop, Genetic and Evolutionary Computation Conference Proceedings*, Springer-Verlag 2004.
- [11] B.J. Ford, “Brownian movement in Clarkia Pollen: A Reprise of the First Observations,” *The Microscope*, volume 40, 4<sup>th</sup> quarter, pp.235-241, Chicago, Illinois, 1992.
- [12] P.E. Merloti, “Ants Box Simulator”, <http://www.merlotti.com/EngHome/Computing/AntsSim/ants.htm>, 2004.

**Paulo Eduardo Merloti** received his BS degree in Data Processing in 1996 from Universidade São Judas Tadeu, São Paulo, Brazil and is currently a graduate student of the Computer Science department at San Diego State University, California. He is a software engineer working for Konexx (San Diego, California). Since 1990 his major passion is to create software. He has been involved with professional software development since 1994, working for companies in Brazil and United States. His research interests include complex adaptive systems, emergent systems, cognitive robotics, machine learning, computer vision, and pattern recognition.

**Joseph Lewis** received his PhD in Computer Science in 2001 from the University of New Mexico, Albuquerque, NM, USA and is currently an Assistant Professor of the Computer Science department at San Diego State University, California. He directs the Science of Evolving and Adaptive Systems (SEAS) Lab with about a dozen graduate students. The primary project is Starcat, a computational framework that generalizes the Copycat family of architectures of Hofstadter and Mitchell. Among the domains this work targets are analogies, robotics, and artificial immune systems.