Percolation Analyses in a Swarm Based Algorithm for Shortest-path Finding

Bruno Panerai Velloso
Federal University of Santa Catarina - UFSC
Campus Trindade
POBox: 476 - Florianópolis - SC - Brasil
bpvelloso@gmail.com

Mauro Roisemberg
Federal University of Santa Catarina - UFSC
Campus Trindade
POBox: 476 - Florianópolis - SC - Brasil
mauro@inf.ufsc.br

ABSTRACT

In this paper we show that the convergence in the Ant Colony Optimization (ACO) algorithm can be described as a *"phase- transition"* phenomenon. The analysis of the ACO with the Percolation Theory approach includes: the pheromone evaporation and the number of agents parameters, so, for a given routing environment, it is possible to select these parameters in order to ensure convergence and to avoid overhead in the algorithm. The objective of this work is to present some experiments that support our hypothesis and to show the methodology used to correlate some algorithm parameters and how they influence in its general performance.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search—*Graph and tree search strategies*

General Terms

Algorithms, Performance.

Keywords

Percolation; Ant Colony Optimization; Swarm Intelligence; Agents behavior.

1. INTRODUCTION

Recent years have seen a growing of application of computational methods based upon natural phenomena with biologically inspired techniques. The ability of those kinds of algorithms to solve complex problems make them attractive as research subjects[6]. In this work we analyze the field of Ant Colony Optimization (ACO), where a model of collective intelligence of ants, using agents of small complexity and without communication, are transformed into useful optimization techniques that can be used in shortest-path finding applications[2]. Because its simplicity and functionality this algorithm is used in many applications as: mobile robots, network routing, network management, transport systems and others[7]. The application described in [5] use Ant Colony Optimization in a mobile phone network routing. In this kind of application the amount of agents is critical and closer optimum values are needed.

Despite its simplicity, the convergence and the mitigating stagnation are the major problems in many ACO algorithms. The proper functioning of the algorithm is highly dependent of a lot of parameters as: evaporation rate, number of agents, aging, pheromone smoothing and limiting, pheromone-heuristic control, and others.

When analyzed with respect the capacity to find a path between two points in a given environment, we can observe that the convergence of the algorithm is highly dependent of the evaporation rate and the number of agents in the environment. If the evaporation rate is too high or the number of agents is too low, the algorithm will not converge and a path between two given points will never be found. If a large number of agents is used, this will generate an unnecessary overhead in the algorithm. So, path finding in ACO seems to be a critical, or phase-transition phenomena that can be modeled with a well established theoretic tool, as is the Percolation Theory.

The first studies about Percolation Theory had been made in the 40s by Flory and Stockmayer [8] and were related with the chemical molecule formation. Later, a critical phenomena approaching was presented. Critical phenomena are classically related to the phase transition, in statistic studies on physics and chemistry and more recently the study of the percolation characteristics he has been successfully used in computational systems and algorithm performance analysis[7].

The objective of this work is to demonstrate the use of the Percolation Theory in Swarm Intelligence algorithms, specifically the ACO algorithm and its variations, in order to analyze its behavior and advice the user about the proper setting of its parameters.

2. RELATED WORK

2.1 Percolation Theory

Percolation is related to the draining of a substance in a random environment. This process can be exemplified by water being drained by a porous material. When water is spilled over a porous material, this material will have its superficial pores filled by water. As these pores are connected

SAC'08 March 16-20, 2008, Fortaleza, Ceará, Brazil

to the internal pores in a greater or lesser number, based in the permeability of the material, the liquid can eventually be lead to the interior and to the other face of the material. When this event occurs it is called percolation, in other words, the fluid percolated the material.

A statistical model of the percolation phenomenon can be made on basis of a graph where each vertex represents a pore, or node, and each edge a path binding two pores, in such a way each vertex present two possible states, "taken" (occupied) or "free" (vacant). The number of "taken" vertices is given by a probability P, according with the material permeability, described as:

Be R a random uniformly distributed variable between [0.1] and P the probability of occupation of each node, we have:

$$R < P, \text{occupied node;}$$
$$R > P, \text{vacant node.}$$

A group of interconnected occupied nodes is called a cluster, or geometric cluster as its generation takes into account only geometric characteristics. As the nodes of a cluster are interconnected by the graph edges, we can see that over a given value of P the cluster can transverse the extremities of the graph. In such cases, the cluster is called a *Percolative Cluster* and represents the possibility of the fluid to percolate the given material, as show in Figure 1.

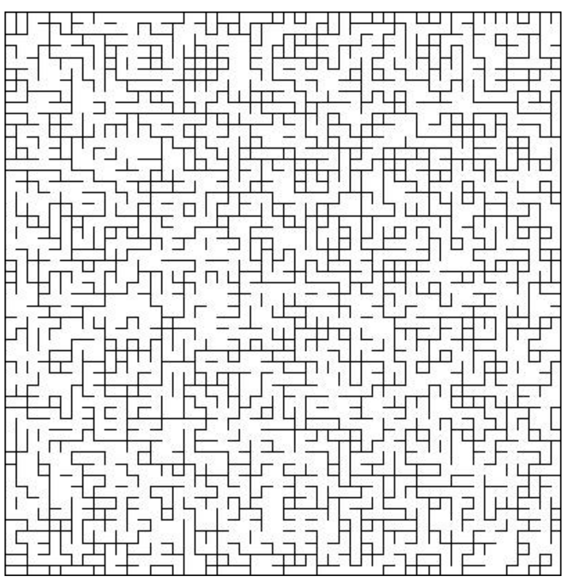

Figure 1: Percolative Clusters - Connections between Nodes

The analysis of critical phenomena in Percolation Theory considers that probability Pc represents the critical probability above of which the existence of a Percolative Cluster [8] as granted as can be seen in Fig. 2. It is said that for $P < Pc$ percolation doesnt occur and for $P > Pc$ we always have percolation. Equation 1 describes de existence of the percolation phenomenon.

$$Percolation(x) = \begin{cases} 0, P < Pc \\ 1, P < Pc \end{cases} \tag{1}$$

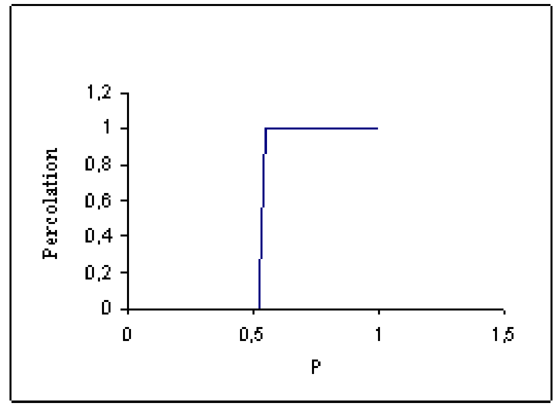

Figure 2: Percolation - Pc≈0.55

According with [4], the critical percolation probability Pc is defined only for infinite length systems. In finite dimension systems this point does not exist, but instead we have is an interval called *Critical Zone*, where the percolation existence is uncertain and vary as function of P. The magnitude of the Critical Zone can be calculated by Equation 2.

$$\delta(N) = \frac{C}{N^{1/2v}} \tag{2}$$

Equation 2 presents the mean square deviation as a function of the amount of nodes N for the percolation in finite dimensions. Here v is called correlation index radius and, for two dimensional problems, has value 1.33 approximately [4]. Constant C in equation 2 depends on the geometric characteristics of the problem, as the distance between nodes and environment area. Fig. 3 presents the expected Percolation Critical Zone for a finite system

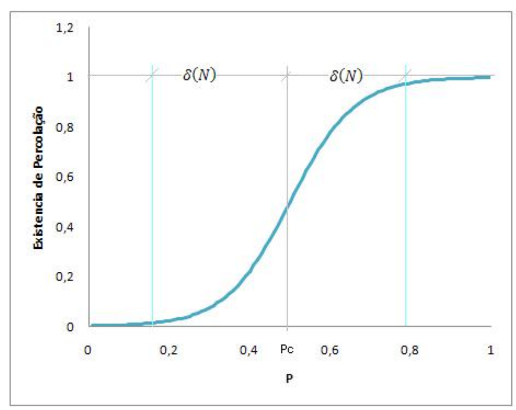

Figure 3: Finite Space Percolation Critical Zone

Analyzing Equation 2 we verify that the width of the Critical Zone is inversely proportional to the number of nodes, approaching to 0 when $N \to 0$ as waited.

Based on equations 1 and 2 we can see that:

$$Pc + \delta(N) \to 1$$
$$Pc - \delta(N) \to 0$$

Nothing can be said about the occurrence of percolation inside the critical zone [4].

2.2 Ant Colony Optimization

The Ant Colony Optimization(ACO) algorithm is a Swarm Intelligence problem-solving computing technique inspired by social insects such as ants. The idea is based in the collective behavior that emerges when a simple ant share information with other ants through stigmergy, an indirect form of communication. The stigmergic communication is achieved by laying a chemical substance called pheromone in the environment which can be sensed by other ants [5]. A colony of ants can perform useful tasks such as building nests, and foraging (searching for food) finding the shortest path to the food source [2] [3].

At the beginning all the ants are located in the nest and cover random ways in order to locate a food source. When one of the ants locates a food source its returns to the nest carrying a load of food, and leaving in nodes where it passes a pheromone track that can be detected by other ants, guiding other individuals to the food place.

The pheromone in its turn presents evaporation as a time function, and the constant replacement is made by the subsequent ants, in such way creating a stable track. When at any point more than one pheromone track is present, the ant has a higher probability to select the track where the pheromone concentration is more intense, what leads, in the convergence of the algorithm, to finding the shortest path between the nest and the food [1].

The convergence of the algorithm involves the appropriate selection of a number of parameters, whose most critical are Number of Agents (Ants) and Pheromone Evaporation Rate. Usually these parameters are established by empirical methods and have direct influence in the performance and overhead of the algorithm.

3. METHODOLOGY

The methodology used in this research consists in the implementation of a simulator for ACO algorithm and collecting statistic data about the convergence of the algorithm when varying parameters. In this case, by convergence we mean the establishment of a continuous pheromone path between the food source and the nest.

Some rules had been taken for the simulator construction:

- The evaporation rate must of such order that a single agent is not capable to produce a continuous path between the food and the nest. In the simulations it is set as fraction of the half-diagonal line of the environment;
- The environment graph is square and it does not have any obstacles;
- Percolation occurs when a continuous pheromone path between food and nest is recognized;
- The number of simulation cycles must be enough to depletes the food source without occurrence of percolation;
- Each set of parameters must be tested many times in order to minimize the statistical errors and distortions relative the use of the finite dimension space with discrete elements.

Regarding the agents a series of behaviors was implemented:

- Each agent start its life cycle in the nest;
- The initial movement is randomly selected, but each agent has a higher probability to move away from the nest. This is necessary because if the movement is purely random, the dislocation tends to zero;
- The agent has the necessary time to cover the complete perimeter of the environment. If after this time it dont locate food or pheromone, the agent returns to the nest and starts a new cycle;
- Whenever an agent crosses a pheromone track they will follow the track in the opposite direction of the nest;
- When an agent finds food they returns in direction to the nest;
- Each agent removes a small part of the food amount;
- Each agent has sensitivity for pheromone in its direct neighborhood, 8-Neighborhood.

Some characteristics of the agents must be remarked: each agent has the knowledge of the nest location; the agent do not have knowledge of other agents; the agent does not has memory and has purely reactive behavior.

The localization of the nest is fixed in all simulations in the environment center, the food source have random localization but, with nest proximity restrictions, being able to be located only near the environment corners.

3.1 Simulations

In this research a series of simulations were performed. In each simulation series, the size of the environment is fixed and the pheromone Evaporation Rate is determined considering the "one agent path restriction" already discussed. Each simulation cycle can be finished by two events: when percolation occurs (a path between food and nest was found), or the maximum number of iterations is reached.

Each simulation cycle is represented by a number of iterations to the complete execution of the ACO algorithm looking for the occurrence of the percolation. In each iteration the state of all the agents is evaluated and updated. For a sequence of simulation cycles a start density of agents relative to the environment area, a final density, and the increment step are established. Each cycle is replicated 10 times in order to minimize statistical errors.

Table 1[1] shows a set of parameters submitted to the simulator. Figure 4 presents the simulation results concerning the probability of occurrence of percolation for different agents density.

4. RESULTS

As a complement of the results shown in Figure 4, we can seen in Figure 5 e 6 other interesting simulation results. Figure 5 presents the number of iterations of the ACO algorithm necessary for percolation (for find a continuous path) and Figure 6 presents the amount of food collected per cycle.

[1]Units Legend: a - area; t - time; f - food; p - pheromone; g - agents

Table 1: Simulation 1

Parameter	Value
Environment Size (L)	100x100 u.a.
Maximum iterations per cycle	10000 u.t.
Available food	10000 u.f.
Start Density (Dini)	0,01 u.g./u.a.
Final Density (Dfim)	0,9 u.g./u.a.
Pheromone deposit (DE)	23 u.p.
Evaporation Rate (TE)	1 u.p./u.t.
Number of cycle per density	50
Density increment step	0,01 u.g./u.a.

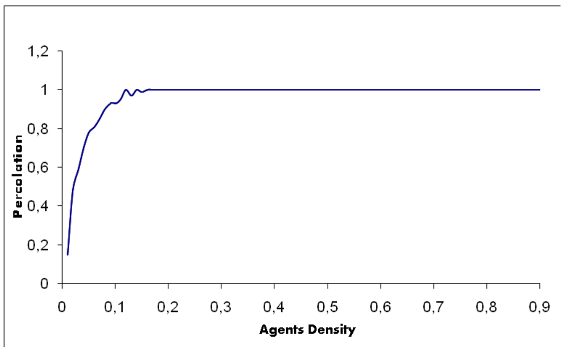

Figure 4: Percolation x Agents Density - Simulation 1

The analysis of Figure 4 clarify the percolation-like behavior of the algorithm and becomes possible to affirm that percolation exists, for the considered parameters set, when the agents density is greater than 0.12 approximately, so:

$$Pc + \delta(10000) = 0.12$$

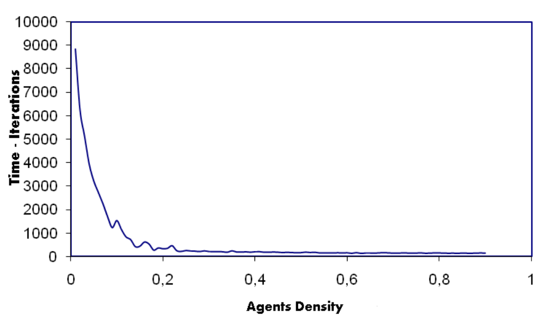

Figure 5: Time in Iterations x Agents Density - Simulation 1

Figure 5 demonstrates that are little improvement in the number of iterations when we increase the agents density after point 0.12. It is important to remember that graph time is represented in iterations and higher agents density demands longer simulation processing time. So we can establish that shorter execution time is located next to point 0.12.

Analyzing Figure 6 we can identify that for densities lower than 0,12, the agents are incapable to form a pheromone

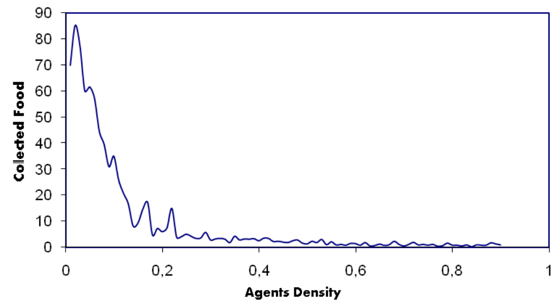

Figure 6: Collected Food x Agents Density - Simulation 1

Table 2: DE Variation

Parameter	Value
Environment Size (L)	100x100 u.a.
Maximum iterations per cycle	10000 u.t.
Available food	10000 u.f.
Start Density (Dini)	0,001 u.g./u.a.
Final Density (Dfim)	0,1 u.g./u.a.
Pheromone deposit (DE)	13 to 32 u.p.
Evaporation Rate (TE)	1 u.p./u.t.
Number of cycle per density	30
Density increment step	0,001 u.g./u.a.

path between the nest and the food even when they localize the food source. The values in this graph are proportional to the number of agents who had located the food.

For the first simulation, density 0.12 represents the percolation critical zone, but this value do not represents a constant for all the possible parameters combination, and is valid just for the parameters presented in Table 1.

The simulation shows, as expected, that the most relevant percolation parameters, for the ACO algorithm are: the environment area; the number of agents; the amount of deposit and evaporation rate of the pheromone. The parameters related to the pheromone can be represent by the maximum length of the pheromone track left by a single agent. As consequence they can be dealt as just one parameter, fixing evaporation rate with value 1 and varying the amount of deposit for each agent.

Figure 7 shows a series of simulation with the parameters set presented in table 2[1] these simulations represent the behavior of the algorithm with the variation in the amount of pheromone deposited by each agent(DE) .

The influence of the DE parameter in width of the percolation critical zone can be perceived. Assuming the Pc of each simulation next to the point where the graph crosses 0.5 in y axis we are able to calculate the values of Pc and δ(N) and with those values it is possible to calculate constant C of Equation 2 as a function of DE. The data are presented in Table 3.

With the values of C we can calculate an approximation in the form of Equation 3:

[1]Units Legend: a - area; t - time; f - food; p - pheromone; g - agents

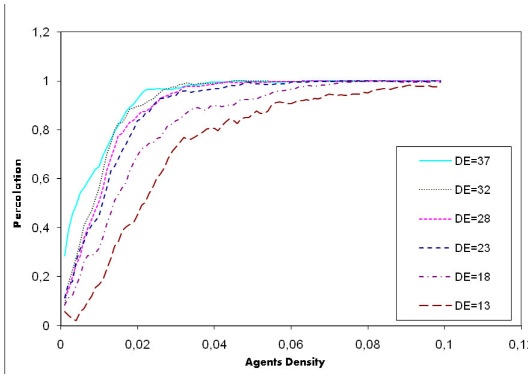

Figure 7: Percolation x Agents Density for DE parameter variation

Table 3: Graph area: 100x100

DE	PC	$PC + \delta(N)$	$\delta(N)$	C
13	0.021	0.112	0.091	2.9026909
18	0.014	0.074	0.06	1.91386213
23	0.011	0.051	0.04	1.27590809
28	0.01	0.043	0.033	1.05262417
32	0.009	0.032	0.023	0.73364715
37	0.004	0.024	0.02	0.63795404

$$C = \frac{29.7}{DE} \tag{3}$$

With equations 2 and 3 we have:

$$\delta_{100x100}(N, DE) = \frac{29.7}{DExN^{1/2v}} \tag{4}$$

Figure 8 shows a comparison between values of constant C obtained from the simulation and the values calculated using Equation 3.

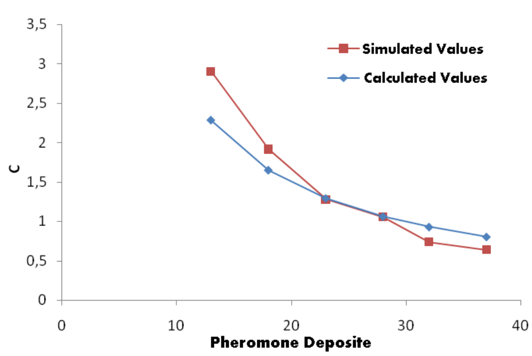

Figure 8: Simulated Values x Calculated Values

The described process was repeated for environments of dimensions 50x50 and 100x100 generating Equations 5 and 6.

$$\delta_{50x50}(N, DE) = \frac{3.8}{DExN^{1/2v}} \tag{5}$$

$$\delta_{150x150}(N, DE) = \frac{206.7}{DExN^{1/2v}} \tag{6}$$

The similarity between Equations 4, 5 and 6 with variation only of the numerical constant make possible to calculate an approximation for the constant as a function of the environment area as shown in Equation 7:

$$\delta(N, DE) = \frac{1.00024^N}{DExN^{1/2v}} \tag{7}$$

Using Equation 7, it is possible to calculate the percolation critical zone width for an ACO algorithm. In order to find the parameters that lead to better resource utilization and guarantees the convergence of the algorithm, we must found the point where percolation initiates in determined application and based in this inferior limit calculate the superior limit of the critical zone.

5. CONCLUSIONS

In this paper we showed that the convergence in the ACO algorithm can be described as a *"phase- transition"* phenomenon that can be modeled with the Percolation Theory. This approach allows better parameters allocation and convergence of the algorithm.

Equation 7 was validated in simulations with interpolated values and had an error better than 0.01 u.g. /u.a., for agents density; however, it expects that this error is grater for values outside the simulations intervals.

In this paper we havent considered the existence of obstacles between the food source and nest. This question is been studied and will be treated in a future work.

6. REFERENCES

[1] H. M. Balch T. Social potentials for scalable multi-robot formations. *IEEE International Conference on Robotics and Automation*, 2000.

[2] T. G. Bonabeau E., Dorigo M. *Swarm Intelligence: From Natural to Artificial Systems*. MIT Press, Cambridge, MA, 1999.

[3] A. R. C. *Behavior-Based Robotics*. MIT Press, Cambridge, MA, 1998.

[4] A. Efros. *Fisica y Geometria del Desorden*. Ed. Mir, Moscow, 1987. traslated. Belosov, S.

[5] W. H. S. Kwang M. S. Ant colony optimization for routing and load-balancing: Survey and new direction. *IEEE Transaction on systems, man, and cybernetics*, Vol 33(No. 5), September 2003.

[6] H. R. Leung H. Phase transition in a swarm algorithm for self-organized construction. *Physical Review*, (E. 68), 2003.

[7] K. R. Leung H. Self-organized construction of spatial structures by swarms of autonomous mobile agents. Master's thesis, College of Engineering. University of Cincinnate., 2003.

[8] A. A. Stauffer D. *Percolation Theory*. Taylor and Francis, London, 1992.