



HAL
open science

Convergence and rate of convergence of a simple ant model

Amine Boumaza, Bruno Scherrer

► **To cite this version:**

Amine Boumaza, Bruno Scherrer. Convergence and rate of convergence of a simple ant model. International Conference on Autonomous Agents and Multiagent Systems - AAMAS'07, May 2007, Honolulu Hawaii, United States. pp.607-609. inria-00170186

HAL Id: inria-00170186

<https://inria.hal.science/inria-00170186>

Submitted on 6 Sep 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Convergence and Rate of Convergence of a Simple Ant Model

Amine Boumaza
LORIA,
Campus Scientifique BP 239
54506 Vandœuvre-lès-Nancy CEDEX FRANCE
boumaza@loria.fr

Bruno Scherrer
LORIA,
Campus Scientifique BP 239
54506 Vandœuvre-lès-Nancy CEDEX FRANCE
scherrer@loria.fr

ABSTRACT

We present a simple ant model that solves a discrete foraging problem. We provide simulations and a convergence analysis. We argue that the ant population computes the solutions of some optimal control problems and converges in some well defined sense. We also discuss the rate of convergence with respect to the number of ants: we give experimental and theoretical arguments that suggest that this rate is superlinear with respect to the number of agents.

General Terms

Algorithms, Theory

Keywords

Multiagent Systems::Emergent behavior, Multiagent Systems::Multiagent planning, Agents::Formal models of agency

1. A SIMPLE ANT MODEL

Consider a set of artificial ants that move on a 2D-grid (the environment) on which they update artificial “pheromone traces”. Each cell s of this grid stores two pheromone traces as two real numbers: $V_1(s)$ and $V_2(s)$. We consider that there are exactly four types of grid cells: one cell is the *nest*, one cell is a *food source*, there are several *bad* cells (or obstacles) and all the others are *free* cells.

Each ant can be in two possible states: either it is carrying food (state “has food”) or it is carrying nothing (state “has nothing”). Ants move around and their state change according to the following natural rules: when an ant gets to the food source, its state is set to “has food”; When an ant gets to the nest, its state is set to “has nothing”; if before being set to “has nothing”, the ant that got to the nest had food (i.e. it has just brought food to the nest) then we increment a food counter. This food counter will measure how much food is foraged by the whole ant population.

We now describe the dynamics of this ant model. At the

beginning: a) the food counter is set to 0, b) the positions of the ants are initialized arbitrarily, c) all ants are set to the “carry nothing” state and d) the pheromone values are initialized arbitrarily. At each time step, each ant does two things:

- It updates the two local pheromone traces $V_1(s)$ and $V_2(s)$ of its current cell s using the pheromone values of its four neighbours. Let us write $\mathcal{N}(s)$ the set of neighbours. For $i \in \{1, 2\}$, let us note $\max^i(\mathcal{N}(s)) \triangleq \max_{s' \in \mathcal{N}(s)} V_i(s')$ and $\text{avg}^i(\mathcal{N}(s)) \triangleq \frac{1}{4} \sum_{s' \in \mathcal{N}(s)} V_i(s')$. The pheromones updates go as follows:

$$V_1(s) \leftarrow \begin{cases} -1 & \text{if } s \text{ is a } \textit{bad} \text{ cell} \\ 1 & \text{if } s \text{ is the } \textit{food source} \text{ cell} \\ \beta (\alpha \max^1(\mathcal{N}(s)) + (1 - \alpha) \text{avg}^1(\mathcal{N}(s))) & \text{otherwise} \end{cases}$$

$$V_2(s) \leftarrow \begin{cases} -1 & \text{if } s \text{ is a } \textit{bad} \text{ cell} \\ 1 & \text{if } s \text{ is the } \textit{nest} \text{ cell} \\ \beta (\alpha \max^2(\mathcal{N}(s)) + (1 - \alpha) \text{avg}^2(\mathcal{N}(s))) & \text{otherwise} \end{cases}$$

where $0 \leq \alpha \leq 1$ and $0 \leq \beta < 1$ with the condition that $\beta < 1$ if $\alpha = 1$.

- It moves to one of its neighboring cell: with probability ϵ ($0 \leq \epsilon \leq 1$) (that we shall call the *exploration* rate) it moves uniformly at random to one of its neighbors. With probability $1 - \epsilon$, it moves to the neighbors that has the highest “pheromone” value V_1 or V_2 depending on its state: the ant uses the pheromone values of V_1 if it is in state “carry nothing” and the values of V_2 if it is in state “carry food”.

The β parameter can be seen as some sort of *evaporation* parameter and is typically set close to 1. The α parameter should typically be set either close to 0 or close to 1. Two simple peculiar instances of our model correspond to the parameter choices ($\alpha = 1, 0 < \beta < 1$) and ($\alpha = 0, \beta = 1$). In the former choice, the general pheromone equation (the “otherwise” case above) reduces to $V_i(s) \leftarrow \beta \max^i(\mathcal{N}(s))$ and in the latter choice to: $V_i(s) \leftarrow \text{avg}^i(\mathcal{N}(s))$ which is a simple linear update. The question whether the ant activities (pheromone values updates and moves) are done synchronously or in some sort of asynchronous way is unimportant.

2. SIMULATIONS

Let us consider a typical run of the algorithm during which we fixed the parameters as follow: $\epsilon = 0.8$, $\alpha = 0.7$, $\beta = 0.9999$ and m the population size is set to 150. Figure 1 shows the evolution of the ants starting at the nest and evolving through the environment searching for food. Once the food source is found by an ant, more ants move toward the source and a trail emerges between the nest and the source. As time goes, a good proportion of the ants follow the trail and the dynamics seems to stabilize. Depending on

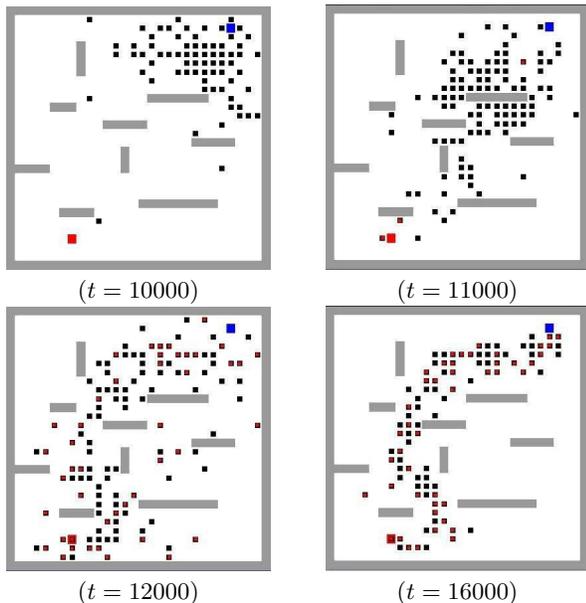


Figure 1: A typical dynamics of the ant model.

the value of α , we can observe different forms of paths, or even sometimes no path. Figure 2 illustrates the asymptotic distribution of ants for different values of α : there are paths except for the case $\alpha = 0.6$ (see [2] for an explanation).

Suppose a path emerges between the nest and the food source. It would be interesting to measure something objective that reveals this path. This can be done through the food counter we introduced earlier. We can draw the curve showing the increase of food brought back to the nest over time. A typical such curve is shown in figure 3. After some time, we observe that this food quantity increases linearly: this means that the foraging behavior of the ants has somehow converged.

The food quantity curve experimentally shows that there is convergence. It can also be exploited to measure a convergence rate. To do so, we fit a line to the linear part of the curve (see figure 3) and define the time of convergence as the intersection of this line with the x-axis. The rate of convergence is then computed as the inverse of the time of convergence. One can compute some statistics on the rate of convergence to evaluate its dependence with respect to the number of ants. Taking $\alpha = 0.7$, $\beta = 0.9999$, $\epsilon = 0.8$, we measured the rate of convergence for various population sizes m . Figure 4 shows the mean convergence rate and its standard deviation out of 20 runs. Such a curve shows exper-

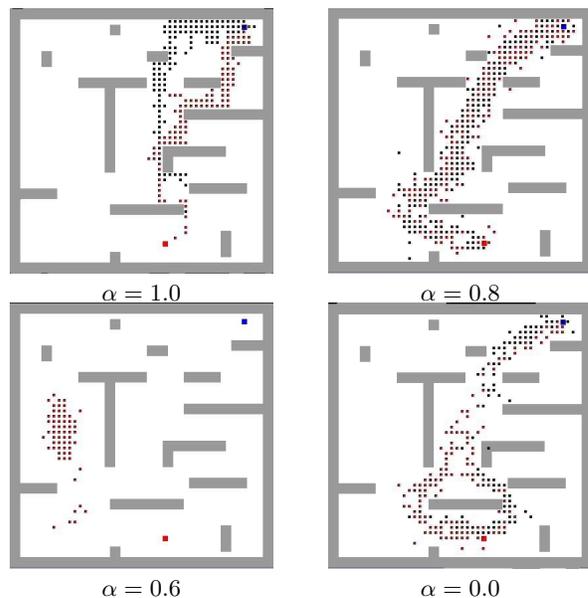


Figure 2: Limit distribution of the ants for various values of α .

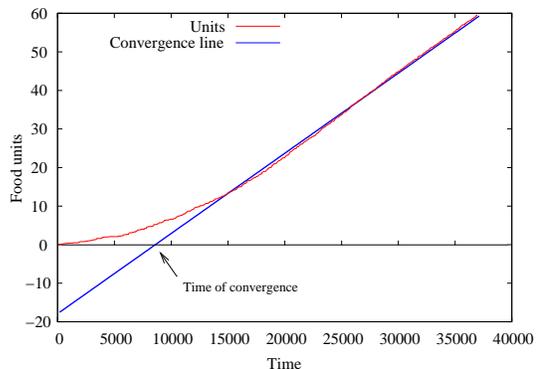


Figure 3: Evolution of the food counter and measure of the time of convergence.

imentally that when the number of ants increases linearly, the increase in rate of convergence is superlinear. In other words, if the number of ants is doubled, the rate increases by more than a factor two.

3. ANALYSIS

The ant algorithm we have just described can be analyzed within the framework of Markov Decision Processes (MDP), that aims to solve optimal control problems [6]. Indeed, it can be shown that the pheromone values V_1 and V_2 each converge to the optimal value function (i.e. the solution) of a control problem:

PROPOSITION 1. *Consider the ant model described in section 1. If the exploration rate $\epsilon > 0$, then the pheromone value V_1 (resp. V_2) asymptotically converges to the optimal value function of the MDP $M_1 = \langle S, A, T_1, R_1 \rangle$ (resp. $M_2 = \langle S, A, T_1, R_1 \rangle$) with discount factor β where:*

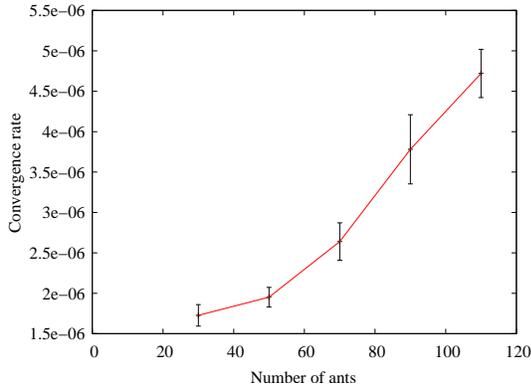


Figure 4: Convergence rate with respect to the number of ants.

- The state space S is the set of grid cells plus an extra “end state”.
- The action space A is the set of cardinal moves (north, east, south, west).
- The transition T_1 (resp. T_2) is characterized as follows: 1) when, in a state s corresponding to a free cell or the nest cell (resp. to a free cell or the food cell), one chooses one of the four directions $a \in A$, the probability of actually making the move in the direction a is $\alpha + \frac{1-\alpha}{4} = \frac{3\alpha+1}{4}$ while the probability of getting to each of the 3 other neighbors is $\frac{1-\alpha}{4}$. 2) From all the other states, that is from bad cells, the food and the “end state” (resp. from bad cells, the nest and the “end state”), there is, for every action, a probability 1 to get to the “end state”, which is an absorbing state.
- The reward R_1 (resp. R_2) is defined as follows: For all states s corresponding to a free cell or a nest cell (resp. a free cell or a food cell), the reward is 0. For the state corresponding to the food (resp. the nest), the reward is 1. The reward is -1 for all states corresponding to bad cells and 0 for the “end state”.

The proof of the above result simply consists in checking that the pheromone updates of our algorithm are equivalent to some asynchronous version of the value iteration algorithm, a standard algorithm for solving optimal control problems [6], for the problems M_1 and M_2 (see [2] for more details). Asymptotically, the moves of the ants, which are a mixture of random uniform moves and the action that climbs up the pheromone value (cf. section 1), correspond to a mixture of random uniform moves and the optimal corresponding moves. Depending on whether an ant is carrying food or not, it uses one pheromone or the other, i.e. it moves toward the nest or the food source. This makes the ants go back and forth between the food source and the nest, and explains the emergence of paths.

It is also possible to study the rate of convergence with respect to the number of ants. Seeing the environment as a graph, and the ants as parallel random walks on it, it can be shown [2] that the convergence time is related to the time necessary for the walks to visit every node of the graph,

which is known as the *cover time of the graph by these parallel random walks* [1]. Though difficult to compute in general, an asymptotic result on cover time [1] allows to state a proposition that sheds light on our observation of superlinear rate of convergence:

PROPOSITION 2. *Consider the ant model described in section 1 on a toric grid environment with n cells, with an exploration rate $\epsilon = 1$ and initialize the ants uniformly on the environment. When the size of the environment $n \rightarrow \infty$ and while the number m of ants satisfies $m \geq 6 \log n$, the time for the pheromone values to reduce their distance to their limit by a factor β is bounded by $\frac{(25+o(1))n^2 \log^2 n}{m^2}$ that is a function that is inversely quadratic in the number m of ants.*

It is known [1] that such a bound is sharp when the environment is a n -cycle graph, that is a long (cycled) corridor: in such a case, the rate of convergence of the pheromone values to their limits is quadratic in the number of ants. Though this result is limited to the case $\epsilon = 1$ (where the ants never exploit the pheromone values), we believe that the above bound remains close when ϵ is close to 1, and this supports our empirical observation of superlinearity.

Our work is close to previous recent works. In [3] the authors tackle foraging problem with multiple pheromone landscapes by exploiting the optimal control theory (in fact they refer to reinforcement learning [6] which is also formulated through MDPs). In [5], the author presents an algorithm that compute shortest path using an asynchronous implementation of the Bellman-Ford algorithm: the local update is of the form $U(x) \leftarrow 1 + \min_{x' \in \mathcal{N}(x)} U(x')$. Modulo the variable change $U \leftrightarrow \frac{\log(V)}{\log(\beta)}$ this is equivalent to our pheromone update with $\alpha = 1$. The main contribution of our work is to deepen the analytical view on these algorithms: if a convergence analysis of [5] can be done [4], [3] only provides experimental evidence. Furthermore, to our knowledge, our work is the only one that tries to tackle the rate of convergence analytically. More details and discussion can be found in an extended version of this article [2].

4. REFERENCES

- [1] D. Aldous and J. Fill. *Reversible Markov Chains and Random Walks on Graphs*. Monograph in preparation, 1996.
- [2] A. Boumaza and B. Scherrer. Convergence and rate of convergence of a simple ant algorithm. Inria technical report 00121341, 2006.
- [3] L. Panait and S. Luke. A pheromone-based utility model for collaborative foraging. In *International Joint Conference on Autonomous Agents and Multiagent Systems*, 2004.
- [4] O. Simonin. Personal communication.
- [5] O. Simonin. Construction of numerical potential fields with reactive agents. In *International Joint Conference on Autonomous Agents and Multiagent Systems*, 2005.
- [6] R. S. Sutton and A. G. Barto. *Reinforcement Learning : An Introduction*. MIT Press, 1998.