

4 Some applications of distributed intelligence: ant algorithms

There is some degree of communication among the ants, just enough to keep them from wandering off completely at random. By this minimal communication they can remind each other that they are not alone but are cooperating with teammates.

D. Hofstadter (1979)

In the previous chapter we looked at a number of examples of distributed systems. Among others, we saw that ants are capable of finding the shortest path to a food source via a process of self-organization mediated through pheromone trails (Chapter 3, Section 3.2). Through such stigmergic interactions¹, i.e. interactions mediated by modifications of the environment (deposition of pheromones), ants manage to solve a complex optimization problem. In addition to solving such a problem, their behavior is robust, i.e. the “solution” is immune to noise. The ants remain also adaptive, in the sense that if the environmental situation changes, for example if a path is no longer present or a shortcut becomes available, they find alternative solutions. These characteristics have led to the development of artificial systems, which were inspired by the observation of colonies of social insects in nature. These systems consist of several agents (e.g. artificial ants) with simple basic capabilities, which give rise to a highly complex structure as a resulting emergent behavior – for good overviews see Bonabeau et al., 2000 or Kennedy and Eberhart (2001).

4.1 Ant-based control

The remarkable properties of ants have encouraged researchers at Hewlett-Packard and British Telecom to try to apply ant-inspired “control rules” to the problem of load balancing and message routing in telecommunications networks. Load balancing is essentially the construction of phone-call routing schemes that distribute the changing load over the system and minimize lost calls. Lost calls are those that never reach their destination (when the caller gets only a beep signal). Their network model is populated by agents (artificial ants), which make use of the trail-laying principles, i.e. they deposit pheromone on each node they visit during their trip through the network. The routing of calls is then decided based on the distribution of pheromone.

Schoonderwoerd, Holland, Bruten, and Rothkrantz (1997) developed the following basic idea. Electronic ants are continuously generated at any node in the network and are assigned random destination nodes. On

¹ Stigmergy is a form of indirect communication through the environment, either by physically changing, or by depositing something on the environment.

their way to the respective destination node ants move around in the network and leave their (electronic) “pheromone trails”. They do this by updating the so-called pheromone tables in the routers (routing tables). Every node has a pheromone table for every possible destination in the network and the destinations’ neighboring nodes. Thus a node with k neighbors in a network with n nodes has a pheromone table with $(n-1)$ rows, where each row has k entries of which each one corresponds to a destination node. The pheromone table contains probabilities (representing the amount of pheromone), which are regularly updated as soon as an ant reaches a node (see Figure 1 below). Updating the probabilities is thus equivalent to the secretion of a pheromone.

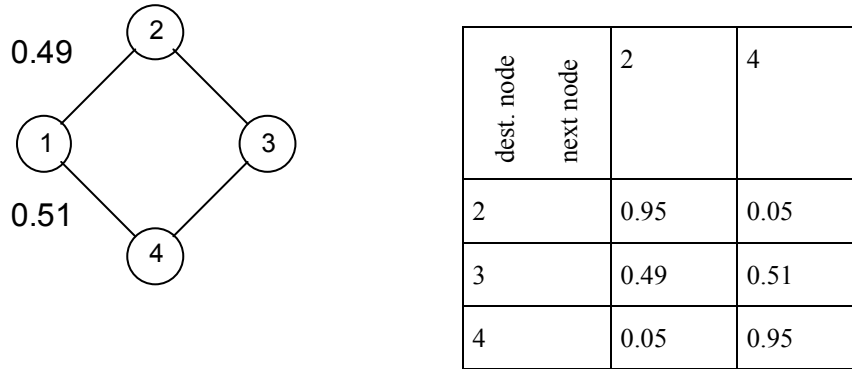


Figure 1: A simple network configuration and the corresponding pheromone table for node 1. Ants traveling from node 1 to node 3 have a 0.49 probability to choose node 2 and a 0.51 probability to choose node 4 as their next node.

Since ants at every step have accurate recent information about their trip from the source node to their current node, the entries in the pheromone tables are updated with reference to the source node. Thus ants directly influence those ants traveling towards their source node and only indirectly those traveling in the same direction.

The probabilities p in the pheromone table are updated according to two formulas. The entry corresponding to the node from which the ant just came is increased according to:

$$p_{new} = \frac{p_{old} + \Delta p}{1 + \Delta p}$$

whereas all other entries are decreased

$$p_{new} = \frac{p_{old}}{1 + \Delta p}.$$

Here Δp is the probability change given by the age of the ants (see below). Note that the influence of a given Δp is much greater on small rather than large probabilities (p_{old}). Thus the entries of rarely used nodes – those nodes with small probabilities – increase faster if traversed by ants.

In order to distinguish the length of the different routes taken, ants get older with every time step while moving along the network or while being delayed at congested nodes. The older an ant gets the smaller the amount of pheromone it is able to lay and thus the smaller its influence on the update of the pheromone

tables. The value Δp used to change the entries in the pheromone tables is reduced in accordance with the age of an ant.

Ants knowing their source and destination node choose their path according to the probabilities stated in the pheromone tables. Further details of how this is done, are given in the paper. The pheromone tables are then used to route an incoming phone call. Note that calls operate independently of the ants. The route of an incoming call is decided according to the probabilities in the pheromone table. The node with the largest probability will usually be the next node visited on its way to its destination node. Calls influence the loads on the nodes and thus introduce delays, and thereby influence the age of the ants visiting the same node. This leads to a complex interaction between the electronic ants and the calls that are routed over the network.

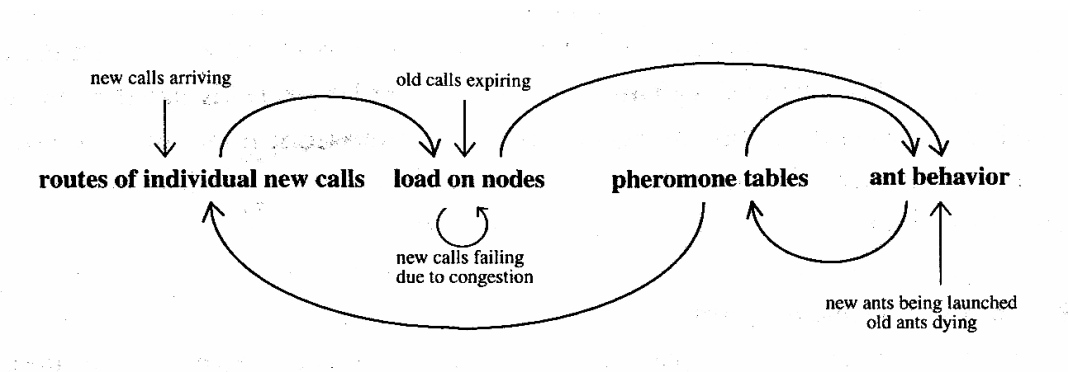


Figure 2: Relationship between calls, node utilization, pheromone table, and ants. An arrow indicates the direction of influence.

The pheromone mechanism of the natural ants had to be changed in order to have different pheromones in the routing tables for each destination node. When trying to map ideas from nature to technology, it is normal that nature's solutions cannot be applied directly, but need to be appropriately modified. The authors also compare the ant-based control method to other methods (shortest path routing, and a software agent based method). Although there are some tradeoffs, ant-based control scores well in these comparisons. One problem with ant-based control is that whenever there is a significant change in the network, it takes some time before the ants "discover" and mark the new routes with pheromones.

The details can be found in (Schoonderwoerd et al., 1997).

4.2 Ant algorithms for optimization problems

Based on the way ant colonies work, new algorithms, called ant algorithms or ant colony optimization algorithms have been developed. These algorithms are especially suited for finding solutions to difficult discrete optimization problems. A colony of artificial ants cooperates to find good solutions, which are an emergent property of the ants' cooperative interaction. Based on their similarities with ant colonies in nature, ant algorithms are adaptive and robust and can be applied to different versions of the same problem as well as to different optimization problems.

The main traits of artificial ants are taken from their natural model – they are: (1) artificial ants exist in colonies of cooperating individuals; (2) they communicate indirectly by depositing (artificial) pheromone (stigmergic communication); (3) they use a sequence of local moves to find the shortest path from a starting, to a destination point (i.e. the optimal solution to a given problem); and (4) they apply a stochastic decision policy using local information only (i.e. they do not look ahead) to find the best solution. If necessary in order to solve a particular optimization problem, artificial ants can be enriched with additional capabilities not present in their natural counterparts.

In ant systems (ant algorithms) an ant colony of finite size searches collectively for a good solution to a given optimization problem. Each individual ant can find a solution or at least part of a solution to the optimization problem on its own but only when many ants work together can they find the optimal solution. Since the optimal solution can only be found through the global cooperation of all the ants in a colony, it is an emergent result of such this cooperation. While searching for a solution the ants do not communicate directly but indirectly by adding pheromone to the environment. Based on the specific problem, an ant is given a starting state and moves through a sequence of neighboring states trying to find the shortest path. It moves applying a stochastic local search policy directed by its internal state (private information), the pheromone-trails, and information encoded in the environment (together: public information). Ants use this private and public information in order to decide when and where to deposit pheromones. In most applications the amount of pheromone deposited is proportional to the quality of the move an ant has made. Thus the more pheromone, the better the solution found. After an ant has found a solution, it dies, i.e. it is deleted (terminated) from the system.

Applications of such ant algorithms can be divided into two classes: ant algorithms for static, and ant algorithms for dynamic combinatorial optimization problems. In static problems the key-points of the problem are defined at the beginning and do not change while the problem is being solved. In dynamic problems the problem changes as a function of itself, thus the algorithms used to solve such problems must be able to adapt “online” to the changes.

Examples of applications to the first class of problems, i.e. to static combinatorial optimization problems are:

- (1) Traveling Salesman Problem: In this problem, a salesman must find the shortest route while visiting a given number of cities, each city exactly once.
- (2) Quadratic Assignment Problem: Problem of assigning n facilities to n locations so that the costs of the assignment are minimized.
- (3) Job-Shop Scheduling Problem: where – given a set of machines and a set of jobs - operations must be assigned to time intervals in such a way that no two jobs are processed at the same time on the same machine and the maximum of the completion times of all operations is minimized.
- (4) Vehicle Routing Problem: In this problem, the objective is to find minimum cost vehicle routes such that (a) every customer is visited exactly once by exactly one vehicle, (b) for every vehicle the total demand does not exceed the vehicle capacity, (c) the total tour length of each vehicle

does not exceed a given limit, and (d) every vehicle starts and ends its tour in the same position (the depot).

- (5) Shortest Common Supersequence Problem: Given a set of strings over an alphabet, a string of minimal length that is a supersequence of each string of the given set has to be found (a supersequence S of string A can be obtained from A by inserting zero or more characters in A).
- (6) Graph-Coloring Problem: This is the problem of coloring a graph so that the number of colors used is minimal but no elements of the same color are adjacent.
- (7) Sequential Ordering Problem, which consists of finding a minimum weight Hamiltonian path² on a directed graph with weights on the arcs and on the nodes, subject to precedent constraints among the nodes.

The main focus of applications to the second class of problems, to dynamic combinatorial optimization problems is on communication networks, in particular on routing problems. Routing answers the question of how to direct data traffic (e.g. phone calls) through a network, i.e. which node to choose next for a data packet entering the network. Routing mainly consists of building, using and updating routing-tables.

Implementations for communication networks can be divided in two classes

- (a) connection-oriented network routing, where all packets of the same session follow the same path selected by a preliminary setup phase (see ant-based control algorithm described in Section 4.1 above);
- (b) connectionless network routing where data packets of the same session can follow different paths (Internet-type networks).

More details on ant algorithms can be found in (Dorigo et al., 1999).

4.3 Conclusion

Ant Algorithms are good examples of the application of swarm intelligence. They show that algorithms inspired by the observation and application of basic principles of a particular natural phenomenon can lead to the discovery of good solutions to diverse optimization tasks. One of the main characteristics of these algorithms is the fact that good solutions are an emergent property of the cooperative interaction of simple agents. Another characteristic is the indirect (stigmergic) communication (indirect communication mediated by changes in the environment) used by the agents.

Beside the examples shown above, there are a variety of current research efforts using swarm-intelligence based approaches. Insect swarms are – among others - studied to devise different techniques for controlling groups of robots which have to cooperatively transport heavy goods, to find more efficient methods to assign jobs in factories, to solve manufacturing problems, to find information over the World Wide Web and in other large networks and to analyze financial data.

² A Hamiltonian path is a path in which every node is visited only once (see Traveling Salesman Problem above).

4.4 Bibliography

- Bonabeau, E., Dorigo, M. and Theraulaz, G. (2000). Inspiration for optimization from social insect behaviour. *Nature*, 406:39-42.
- Dorigo, M., Di Caro, G. and Gambarella, L.M. (1999). Ant algorithms for discrete optimization. *Proc. of Artificial Life 5*, MIT Press: Cambridge, MA, pp. 137-172.
- Dorigo, M., Maniezzo, V. and Coloni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics*, 26(1):29-41.
- Kennedy, J. and Eberhart, R.C (2001). *Swarm Intelligence*. Morgan Kaufmann Publishers: San Francisco.
- Schoonderwoerd, R., Holland, O.E.; Bruten, J.L. and Rothkrantz, L.J.M. (1997). Ant-based load balancing in telecommunications networks. *Adaptive Behavior*, 5(2):169-207.