

Congestion Control in Ant Like Moving Agent Systems

Alexander Scheidler, Daniel Merkle, and Martin Middendorf

Abstract In this paper we study the problem of congestion in system where agents move according to simple ant inspired movement rules. It is assumed that the agents have to visit a service station to refill their energy storage. After visiting the service station the ants can move randomly and fast. The less energy an agent has the slower it becomes and the more it moves in direction of the service station. Different methods for self-organized congestion control are proposed in this paper where the behavior of the agents compared to the original is not changed or is changed only slightly without the need to use any global information and without using additional sensory information. The proposed systems are investigated with

1 Introduction

The (movement) behavior of social insects is an inspiring source of ideas for the design of methods for solving various problems in computer science and related fields. Examples are the well-known Ant Colony Optimization method as introduced in [1] (for an overview see [2]), algorithms for the movement behavior of robots [3], ant inspired clustering methods (for an overview see [4]), or the self-organized behavior of the compounds in Organic Computing (OC) [5, 6] systems (for an overview see [7]). Emergent pattern

Alexander Scheidler and Martin Middendorf
Department of Computer Science
University of Leipzig, Postfach 100920, 04009 Leipzig, Germany
{scheidler,middendorf}@informatik.uni-leipzig.de

Daniel Merkle
Department of Mathematics & Computer Science
University of Southern Denmark, Campusvej 55, DK-5230 Odense M, Denmark
merkle@imada.sdu.dk

that might arise when groups of animals move have been deeply investigated in biology (e.g., [8, 9, 10, 11]). In [12] the so called sorting behavior in the brood chambers of the ant *Leptothorax unifasciatus* has been investigated. In the brood chambers the youngest brood items (eggs and microlarvae) are placed in the chambers center, larger larvae are arranged in concentric rings around the center, and the largest and oldest brood (pupae and prepupae) is placed in an intermediate area between the peripheral larvae and the larvae of medium size. One explanation why this sorting occurred is that the brood distribution pattern helps to organize the brood care in the nest. It has been shown by simulation studies that a system with very simple behavioral rules for the movement of agents can show an emergent sorting behavior. The sorting behavior of ants has inspired the design of methods that are used by robots to solve sorting problems [13].

In [14] movement models that are inspired by the ant *Leptothorax unifasciatus* ants have been applied to OC systems with moving agents. In the studied OC systems the agents have one (or several) service stations which they have to visit from time to time (e.g., to recharge their batteries or to drop items they have collected). It was shown that emergent patterns that occur within the distribution of the agents with respect to the different service stations can occur even when only very slight behavioral differences between the agents exist. It was also shown that a problem with the ant inspired movement models is that unwanted congestion can emerge at the service stations unless there is only a small number of agents in the system.

In this paper we propose and study some methods to reduce and control the emergent congestion in ant like moving agent systems. The aim is to develop methods where the behavior of the agents is not changed or is changed only slightly without the need to use any global information and without using additional sensory information. While reducing congestion (and thereby increasing the performance of the system) it is important that the fairness of the system is not reduced. Fairness is measured here as the variance of the waiting times of the agents before they can visit the service station.

The paper is structured as follows. In the next Section 2 the agent model and the movement behaviour are introduced. The methods for congestion control are introduced in Section 3. The experiments are described in Section 4. Results are presented in Section 5. Conclusions are given in Section 6

2 Agent Model

Different models for the movement of ants within a nest have been introduced in [12]. It was shown that small differences in the movement behavior of the ants can lead to spatial sorting of the ants (i.e., on average over time ants with different behaviour can be found in different areas of the nest). The degree of the sorting depends on the particular movement model. In principle each

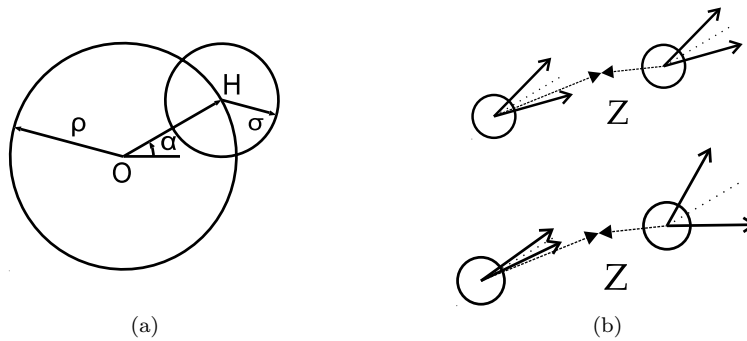


Fig. 1 (a) Every agent is modeled as a disc; ρ - radius; O - centre of the body (x, y) ; α - direction of movement; H - centre of the head; σ - sensing range. (b) Effect of different values of the parameter μ_i on the turning behavior when unobstructed; Z is the service point; (upper) for large μ_i there is only a slight difference between moving from or to the service point; (bottom) for small μ_i the turning angle becomes significantly smaller the smaller the angle between actual moving direction and the vector to the service point is

of the defined movement models consists of the following two actions for an ant: “turning” the movement direction and “moving” straight forward.

In [14] some of the movement models of [12] have been changed slightly to avoid an unnatural blocking effect that was observed and also to adapt the models to fit the requirements of OC systems. It was shown that emergent spatial sorting patterns for groups of randomly moving ant like agents can depend on slight differences in the movement models. It was observed that the movement of ants can depend on the CO_2 gradient which typically points to the center of a brood chamber ([15, 16]). It was argued in [14] that the movement behaviour of agents in OC systems might similarly depend on their movement direction with respect to the direction of a service station. But, different from the natural systems in the OC system there might be several service points that influence the movement of the agents. It was shown that the relative size of the influence area of the service points can lead to an interesting and strong emergent pattern within the spatial distribution of agents with slightly different moving behavior.

The movement models of the agents that are used in this paper correspond to the *repulsive ant model* from [14]. This model which is a mixture of the *centripedal ant model* and the *avoiding ant model* from [12]. This mixed movement model was introduced to overcome the problem that agents get stuck near the focal point (see [14] for details). The behavioural differences between the agents were modeled in the way that each agent i has a parameter $0 \leq \mu_i \leq 1$ that influences its moving behaviour. Fixed values $\mu_i = i/(n+1)$ for agent $i \in 1, \dots, n$ were used for the experiments in [12] and [14]. In this paper we investigate a movement model where the parameter μ_i can vary over time for every agent. The movement behaviour is described in the following.

Shape of the Agents. Similar as in [12] the shape of an agent is modeled as a disc with radius ρ . The centre of the disc (x_i, y_i) represents the position of agent i . Each agent has an actual direction of movement α_i , which is measured as the angle relative to the lower border of the simulation area. The point at position $(x_i + \rho \cos \alpha_i, y_i + \rho \sin \alpha_i)$ models the centre of the agents head. From the centre of the head every agent can sense obstacles within a range of distance σ , called sensing range (see Figure 1). Agent i collides with agent j if agent j is within the sensing range of agent i , i.e., when the distance between the centre of the head of agent i and the centre of the body of agent j is smaller than $\sigma + \rho$. Similarly, an agent collides with the nest wall when the euclidian distance between the centre of its head and the wall is less than the sensing distance.

Movement when unobstructed. If there is no obstacle (wall or other agent) within its sensing range an agent will move and turn at each time step. The agent moves distance ν_i in direction α_i , i.e., $x_i \leftarrow x_i + \nu_i \cos \alpha_i$ and $y_i \leftarrow y_i + \nu_i \sin \alpha_i$. The different values ν_i represent different velocities of the agents. The parameter value ν_i depends on the internal parameter μ_i of the agents as follows: $\nu_i = (1 - \mu_i)\nu_s + \mu_i\nu_f$ where parameters ν_s and ν_f , $0 < \nu_s < \nu_f < 1$ denote the slowest and the fastest velocity. Within the same time step the agent also makes a turn by changing its movement direction by $\alpha_i = \alpha_i + \theta_i$, where θ_i depends on the internal parameter μ_i of the agent and ϕ_i is the angle between the actual moving direction α_i and the vector towards the service point. For the calculation the clinotaxis model from [17] is used: $\theta_i \leftarrow p_u(1 - \mu_i)\chi + p_b\mu_i\tau \cdot (1 - \cos(\phi_i))/2$ where $\chi = 15^\circ$, $\tau = 30^\circ$. The values of p_u and p_b are randomly chosen from $\{-1, 1\}$ and determine the direction of turning. The turning behavior depends on ϕ_i and the larger this angle is the stronger the agent will turn. Agents with larger value μ_i will be less affected by their ϕ_i as agent with small μ_i (see Fig. 1 b). Therefore, for agents with small value μ_i the attraction to the service point is stronger than for agents with large value μ_i .

Movement when obstructed. If the wall or another agent is within the sensing range of an agent, the agent will not move, but only make a turn. It avoids the obstacle explicitly by turning into one direction until it can move again. To define the turning direction assume that agent i collides with agent j . The sign of the scalar product between the vector that is perpendicular to the vector of the moving direction of agent i and the vector from the centre of agent i to the centre of agent j determines the direction of turning: $\theta_i \leftarrow \text{sign}((- \sin \alpha_i, \cos \alpha_i) \cdot (x_j - x_i, y_j - y_i))U(0, \Theta_i)$. A collision with the nest wall is handled analogously.

In our model we use the parameter μ_i to model the state of an agent. The higher the value of μ_i is the faster can the agent move. Therefore, if an agent has visited the service area the value of μ_i is increased. The motivation behind this is that in applications the agents might get new power at the service station or has been unloaded at the service station. During movement

of an agent its value μ_i decreases. The motivation is that in applications the agent might use power or pick up items.

3 Congestion Control

To resolve possible congestion of the agents at the service point we introduce three different congestion control methods. The goal of these methods is to resolve the congestion either by leaving the behaviour of the agents unchanged or by changing the behaviour of the agents only slightly but without need for introducing any new type of sensory information or global knowledge. The first two control methods C_P and C_W do not change the agent behaviour and the third method C_D changes only the sensing range of the agents.

The idea of control method C_P is to introduce two parallel walls next to the service station that form a pipe. The idea of this method is that agents that have visited the service station and have a high value μ_i might be able to move away from the service station through the pipe whereas only few of the agents that have a small value μ_i might use the pipe to move to the service station. An example of a pipe can be seen in the left of Figure 3.

Control method C_W is to introduce two additional walls on two sides of the service station. Each wall has a small opening in the middle that is next to the service station. The idea of this method is that slow agents with small value of μ_i might be forced to wait behind a wall and therefore do not block the service station. Hence, the agents that have visited the service station can move away from it. An example for this control method C_W can be seen in the middle of Figure 3.

The third control method C_D is to change the behavior of the agents slightly. Here the sensing range σ_i of agent i depends on the internal parameter μ_i . The sensing range is calculated as follows: $\sigma_i = 2\rho - 1.4\mu_i\rho$. The idea behind this method is that agents with a small value of μ_i that move to the service station have a larger sensing range and therefore leave some space between them when they come next to the service station. This space can be used by the agents that have visited the service station and therefore have a large value μ_i can to move away from the service station.

4 Experiments

The simulation area is a quadratic field with side length 1. At the start of a simulation run the positions of the agents are distributed randomly with uniform distribution over this area. Also the values of the internal parameters μ_i are chosen randomly with uniform distribution between 0 and 1. In the centre of the field there is a circular service area with radius 0.04. The centre

of this area is the service point. If an agents position (i.e., the centre of its body) is within the service area its internal parameter μ_i is set to 1. If agent i moves (e.g. the agent is unobstructed) the value of μ_i is decreased by a fixed value 0.001 until $\mu_i = 0$. Observe that the smaller the value of μ_i is the slower moves the agent and also the higher is the attraction force to the service point.

For the experiments the body of an agent has radius $\rho = 0.01$, the radius of the head is $\sigma = 0.006$. Parameters ν_s and ν_f that denote the slowest and the fastest velocity are set $\nu_s = 0.0006$ and $\nu_f = 0.006$. Per time step parameter μ_i of agent i reduced by 0.001. Different numbers of agents have been used and for each number of agents and each congestion control method each run was repeated 20 times over 10000 time steps each.

5 Results

Figure 2 shows the distribution of the agents after 2000 time steps for different number of agents. It can be seen that a system with 90 agents works without strong congestion at the service station. It can also be seen that agents with small value μ_i (bright color) tend to be close to the service station. Agents with large value μ_i are nearly randomly distributed over the whole field. This is different for a system with 150 agents. Here most nearly all agents can be found close to the service station. The agents with large value μ_i can be found or very near to the service station. They cannot move away because the way is blocked by the agents with small value μ_i that try to move into the service area. As shown later, for this system the agents cannot do much useful work (if that means that the agents should ideally move over the whole field). Altogether, the observed congestion is an unwanted effect of the system that depends on colony size.

Figure 3 shows the distribution of the agents for a system with 150 agents after 2000 time steps using one of the congestion methods $C_{\mathcal{P}}$ and method $C_{\mathcal{W}}$. It can be seen that there is much less congestion by slow agents with small value μ_i within the pipe for method $C_{\mathcal{P}}$ than outside of the pipe next to the service area. It can also be seen that the agents with high value μ_i can move through the pipe.

For method $C_{\mathcal{W}}$ it can be seen that the congestion around the service area is much less compared to the system without congestion control. Agents with high value μ_i can be found in different parts of the filed and not only next to the service area (as it was the case when no congestion control is used).

For method $C_{\mathcal{D}}$ the distribution of the 150 agents after 2000 time steps is show in Figure 4. The figure shows that at least some agents with high value μ_i that have visited the service station can move away from it because the agents with small value μ_i leave some space between each other.

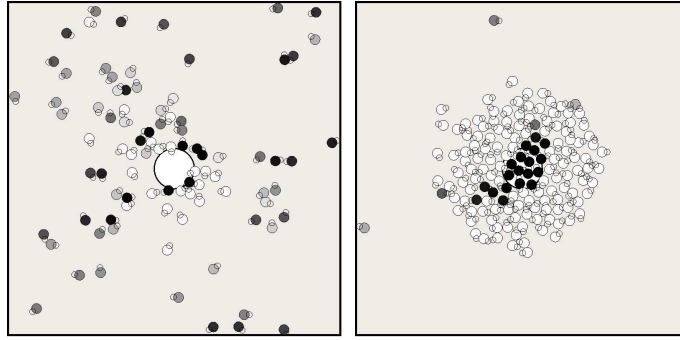


Fig. 2 Distribution of the agents for different number of agents after 2000 time steps; (left) 90 agents; (right) 150 agents; the smaller the value μ_i of an agent the brighter is its color; the service area is the white circle in the middle

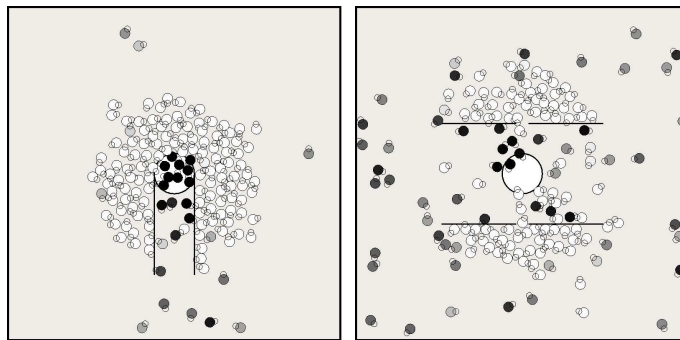


Fig. 3 Distribution of agents for a system with 150 agents after 2000 time steps using congestion method C_P (left), method C_W (right)

To compare the performances of the control methods with system that uses no control method the following measure for the performance of the system is used. If an agent reaches the service area its value μ_i is increased by adding the value $1 - \mu_i$ so that $\mu_i = 1$ holds afterwards. Summing up over all values $1 - \mu_i$ for all i and every time when the value μ_i is increased can be seen as measure for the performance the system. This value is called the total energy consumption of the system and is denoted P_T when measured over the first T time steps. Since agents that move use energy whereas agents that can not move do not use energy the total energy consumption is a measure how freely the agents can move on average.

Figure 5 shows the total energy consumption P_T for a system without congestion control and systems with congestion control. It can be seen that for a small number of agents when no congestion occurs the system without congestion control has the highest performance. This is no surprise because the congestion control methods slightly hinder the agents to move freely within the field when there is no congestion. But when the number of agents

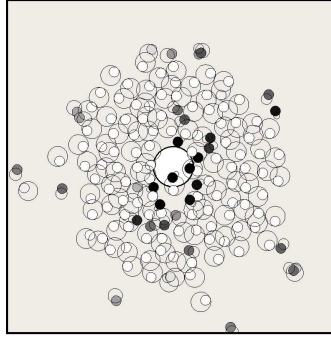


Fig. 4 Distribution of agents for a system with 150 agents after 2000 time steps using a congestion method C_D

becomes larger than 100 the performance of the system without congestion control decreases very fast. For more than 130 agents this system has the worst performance. For a medium number of agents the system with method C_W is the best. But for a large number of agents this method is not much better than a system without congestion control. For a larger number of agents (more than 210) the system with method C_D is clearly the best. Method C_P is better than the system without congestion control for more than 135 agents but it is worse than the other two methods. One reason might be that it is not so easy for the agents that have visited the service station to move away from it through the pipe because they move randomly in the considered model (and do not actively move away from the service station).

Besides the reduction of congestion, fairness for service is another important measure for the collective behavior of agents. In the considered system, e.g., the waiting times for service have to be similar. We measured the fairness of the system in two different ways. Firstly, at the end of a given time interval of length T for every agent the total amount of values that have been added to μ_i for all its visits of the service station is measured. Then the relative standard deviation (RSD) of these values for all agents has been taken as a measure for the fairness of the system (the lower the variance means the more fair the system is).

The behavior of the systems with respect to this fairness measure is shown in the left part of Figure 6. It can be seen that the system without congestion control is most fair for a small number of agents (less than 110 agents). For a larger number of agents the system with the C_D method is the best.

The second measure of fairness is defined as follows. Let $\tau(T)$ be the mean waiting time of the agents where the waiting time of an agent is defined as the length of the time interval from the time when its internal parameter (μ_i) becomes zero until the time when it reached the service area (measured over a simulation run over T time steps). Let $\sigma(T)$ be the standard deviation of these waiting times. A dimensionless measure for the fairness is then defined

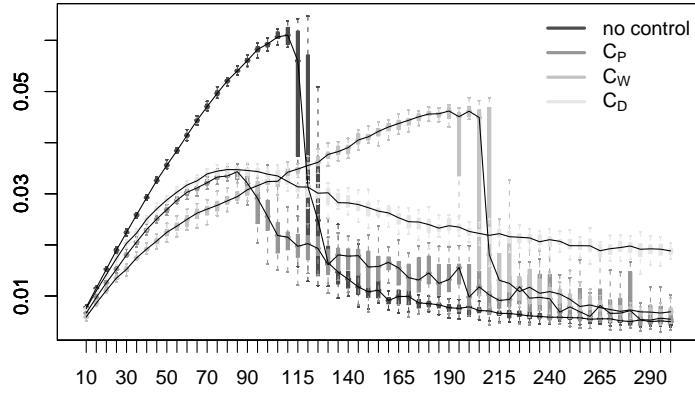


Fig. 5 Total energy consumption P_T for different number of agents measured over 10000 time steps for a system without congestion control and systems with the different congestion control methods

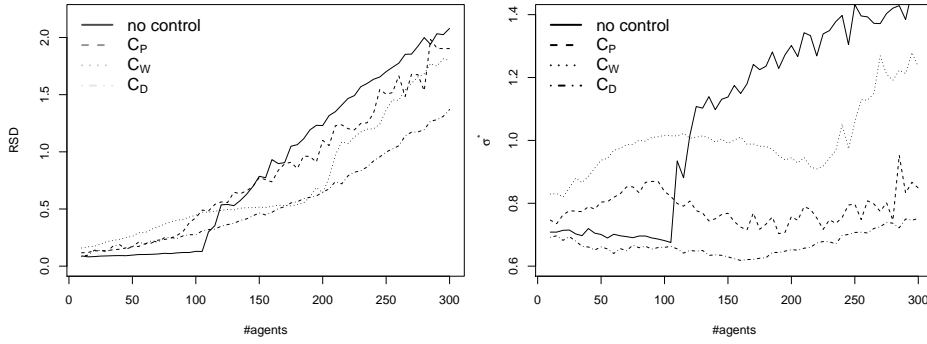


Fig. 6 Fairness for different number of agents measured over 10000 time steps for a system without congestion control and systems with the different congestion control methods; first fairness measure RSD (left), second fairness measure σ^* (right)

similarly as in [?] by $\sigma^*(T) = \sigma(T)/\tau(T)$. Note, that for this measure only the waiting times of the agents that reached the service point are considered. Hence, a congested system may still be fair, if there is only a small subset of agents that are served and these agents have similar waiting times.

The behavior of the systems with respect to the second fairness measure is shown in the right part of Figure 6. It can be seen that the system the C_D method is most fair (independently of the number of agents). For small number of agents (less than 110) the system without congestion control is the second most fair system. For larger number of agents the system with method C_P is the second best.

6 Conclusions

In this paper we have studied the problem of congestion control agent systems with ant inspired movement rules. In the studied systems the agents have to visit a service station to refill their energy storage. Three methods for self-organized congestion control have been proposed. It was shown experimentally that the proposed methods can significantly reduce the congestion and are also fair for systems with a larger number of agents. Differences between the behavior of the systems have been discussed.

Acknowledgment

This work was supported by the German Research Foundation (DFG) through the project Organisation and Control of Self-Organising Systems in Technical Compounds within SPP 1183.

References

1. M. Dorigo, V. Maniezzo, A. Coloni: Positive feedback as a search strategy. *Tech Rep.*, 91-016, Dip Elettronica, Politecnico di Milano, Italy, 1991.
2. M. Dorigo, T. Stützle: Ant Colony Optimization. MIT Press, 2004.
3. J.-L. Deneubourg, S. Goss, N. Franks, A.B. Sendova-Franks, C. Detrain, L. Chretien: The dynamics of collective sorting: Robot-like ants and ant-like robots. In *Proc. of the 1st Int. Conf on Simulation of Adaptive Behavior*, 356–363, 1991.
4. J. Handl, B. Meyer: Ant-based and swarm-based clustering. 1(2), 95-113, 2007
5. H. Schmeck: Organic Computing – A New Vision for Distributed Embedded Systems. Proc. of the Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC 2005), 201-203, 2005.
6. T. Schöler, C. Müller-Schloer: An Observer/Controller Architecture for Adaptive Reconfigurable Stacks. Proceedings ARCS 05, Springer, LNCS 3432, 139-153, 2006.
7. D. Merkle, M. Middendorf, A. Scheidler: Organic Computing and Swarm Intelligence In C. Blum, M. Merkle (Eds.), *Swarm Intelligence*, Springer, 2008.
8. J.L. Deneubourg, A. Lioni, C. Detrain: Dynamics of Aggregation and Emergence of Cooperation. *Biol. Bull.*, 202:262267, 2002.

9. S. Depickère, D. Fresneau, J.-L. Deneubourg. A basis for spatial and social patterns in ant species: dynamics and mechanisms of aggregation. *Journal of Insect Behavior*, 17(1): 81-97, 2004.
10. R.V. Sole, E. Bonabeau, J. Delgado, P. Fernandez, J. Marin: Pattern formation and optimization in army ant raids. *Artificial Life*, 6(3):219-226, 2000.
11. G. Theraulaz, E. Bonabeau, S.C. Nicolis, R.V. Sole, V. Fourcassie, S. Blanco, R. Fournier, J.L. Joly, P. Fernandez, A. Grimal, P. Dalle, J.L. Deneubourg: Spatial patterns in ant colonies. *Proc. Natl. Acad. Sci.*, 99(15):9645-9649, 2002.
12. A.B. Sendova-Franks and J.V. Lent. Random walk models of worker sorting in ant colonies. *Journal of Theoretical Biology*, 217:255-274,2002.
13. C. Melhuish, A. B. Sendova-Franks, S. Scholes, I. Horsfield, F. Welsby: Ant-inspired sorting by robots: the importance of initial clustering. *J R Soc Interface*. 3(7), 235–242, 2006.
14. A. Scheidler, D. Merkle, M. Middendorf: Emergent Sorting Patterns and Individual Differences of Randomly Moving Ant Like Agents. *Proc. 7th German Workshop on Artificial Life (GWAL-7)*, IOS Press, 11 pp., 2006.
15. M.D. Cox, G.B. Blanchard: Gaseous templates in ant nests. *Journal of Theoretic Biology*, 204:223-238, 2000.
16. G. Nicolas and D. Sillans: Immediate and latent effects of carbon dioxide on insects. *Annual Review of Entomology*, 34:97–116, 1989.
17. D. Grünbaum, Schooling as a strategy for taxis in a noisy environment. *Evolutionary Ecology*, 12: 503-522, 1998.