Global Convergence of Local Agent Behaviors

H. Van Dyke Parunak, Sven A. Brueckner, John A. Sauter, Robert Matthews
Altarum Institute
3520 Green Court, Suite 300
Ann Arbor, MI 48105 USA
1-734-302-4600

{van.parunak, sven.brueckner, john.sauter, robert.matthews}@altarum.org

ABSTRACT

Many multi-agent systems seek to reconcile two apparently inconsistent constraints. The system has a global overall objective. However, the agents have only local information to guide their actions. Such systems are presently more art than science. They often exhibit regularities (such as exponential convergence) that we do not understand, and we do not know how to improve their functioning in a disciplined manner. In this paper, we develop a simple statistical model for such systems that can enhance both our intuitions about their functioning and our ability to engineer them, and apply it to three systems that we have constructed.

Categories and Subject Descriptors

F.2.0 [**Analysis of Algorithms**]: General; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *Multiagent Systems*

General Terms

Performance, Experimentation, Theory.

Keywords

Convergence, Emergent Behavior, Clustering, Graph Coloring, Vehicle Routing

1. INTRODUCTION

Many multi-agent systems seek to reconcile two apparently inconsistent constraints. The system's overall objective is defined globally. However, the agents have only local information available to guide their actions. This limitation may be due to many realistic factors, such as lack of long-range communications, proprietary restrictions that limit exchange of information among agents, or the lack of a distinguished agent with sufficient capacity to collect and analyze data from the entire population.

Much current work on constructing systems of this sort is more art than science, relying on a "bag of tricks" whose effectiveness is empirically impressive in specific implementations, but has no guarantee that it can be achieved in systems not yet constructed. To move from tricks to engineering, we must understand the mechanisms by which agents with local knowledge can achieve global ends, and be able to place bounds on how close to a global optimum they can approach and how quickly they can do so.

AAMAS'05, July 2529, 2005, Utrecht, Netherlands.

Empirically, many such systems converge exponentially fast to their global objective. Such convergence is highly desirable, since it supports "any-time" processing, in which the system quickly yields an approximate result and then (if given more time) refines it further. Any convergence curve with a decreasing first derivative would satisfy the "any-time" property, but it is striking that our data frequently show a good fit to an exponential of the form $A \pm Be^{-\lambda t}$. The appearance of this fit in multiple settings suggests that the same underlying principles may be responsible for the behavior of the systems. If we can discover this theory, it would not only guide our scientific intuitions about the behavior of these systems, but also improve our engineering ability to develop and tune such applications.

We can map these systems onto a model of local behavior simple enough to analyze its convergence. This system converges exponentially, as do the more complicated systems to which it is related. Having thus established the relation between our model system and our real systems, we can draw conclusions from the model system that explain other features of the real systems.

Section 2 gives three examples of systems that seek global performance from local interactions. Two of these exhibit exponential convergence, while the third does not. Section 3 presents the "adaptive walk," a model that is simple enough to be analytically tractable but can be mapped onto our examples. We show that this model exhibits exponential convergence and discuss the implications of the convergence equations for designing systems of this type. Section 4 revisits our three examples, explaining both the exponential convergence of the first two and the departure from exponentiality of the third, as well as exploring other of their features in the light of the model system. Section 5 concludes.

2. EMPIRICAL EXAMPLES

Three example systems illustrate global convergence from local agent decisions: distributed clustering inspired by ant nest sorting, distributed graph coloring, and area surveillance using digital pheromones. In each case we motivate the application, identify the agents, the global objective they seek to satisfy, and their local decision algorithm, and describe the system's convergence.

2.1 Ant Clustering [13]

Motivation.—Clustering data items can be a useful foundation for further information retrieval tasks. Most conventional algorithms [5] have two constraints. First, they are *centralized*, requiring access to a central data structure, the similarity matrix, and so are difficult to distribute across many processors as the amount of data to be handled increases. Second, they are *batch*, requiring that both the population of items being clustered and the similarity function being used to compare them remain constant during clustering.

Nature suggests an alternative algorithm that is decentralized and continuous. Ants cluster items in their nests, such as dead ants or food [2]. As they wander about, ants pick up objects with a probability u and drop them with a probability d. The probability u decreases, and d decreases, with the object's similarity to nearby objects. As objects move from regions where they are dissimilar to their surroundings to regions where they are similar, homogeneous clusters form. This algorithm can be distributed over many processors (the ants), and can continue to run as the population of items, or even the similarity function, change.

Agents.—In the adaptation of this algorithm in [13], the objects to be clustered (typically, paragraphs of text) are themselves active "content agents" that move themselves rather than relying on a separate species of agents. Initially, content agents are assigned randomly to a fixed set of logical "places" that may be distributed over multiple processors.

Global Objective.— Each content agent has a binary vector indicating which concepts it attests. The similarity of two content agents is the cosine of the angle between their vectors. Each place's Homogeneity is the mean of the pairwise similarities among its content agents. The global objective is to maximize the average of the place homogeneity across all places.

Local Decisions.—A content agent monitors its similarity to a random sample of other content agents in its place, and to a random sample of other places. It decides to change its place probabilistically, where the probability increases with the increase in similarity that the move would provide. The decision is local in three ways. 1) A content agent considers only a restricted number of places and a limited number of content agents in each place. 2) An agent's greater similarity to a new place than to its current place does not guarantee that its move will increase the homogeneity of either of those places. 3) Other content agents are concurrently making movement decisions, changing the environment as decisions are made. The algorithm selects documents randomly with replacement. Each time a document is selected, it decides whether or not to move.

Convergence.—This algorithm was tested on a subset of 10^7 candidate paragraphs that matched a test concept map. Table 1 shows the parameters for a typical run. Figure 1 shows the convergence of average place homogeneity over the course of the run. The solid line is a least-squares fit to $A - B \cdot e^{-\lambda t}$.

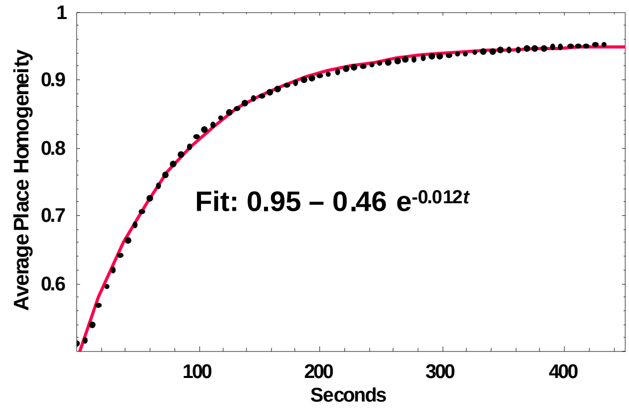

Figure 1. Ant clustering convergence with fit (solid line) fit.

Another experiment explored the system's adaptability by adding new documents at a constant rate to random places as the clustering proceeds. Figure 2 shows that over a wide range of rates, the exponential fit is excellent with a decreasing asymptote. Surprisingly, the exponent λ is essentially constant. Intuitively, as N increases, λ should decrease, since more documents need to be sampled and potentially moved to achieve the same homogeneity.

Table 1. Parameters for Ant Clustering

Length of concept vector	5
Total number of documents	17221
Number of places per processor	100
Number of processors	9
Number of destinations considered	30
Probability of considering a place on another processor	1/30
Max number of docs compared in each place	30

2.2 Graph Coloring [1]

Motivation.—Graph coloring is a useful abstraction of many resource allocation problems. Each node represents a task, each color represents a resource, and an edge between two nodes indicates that a single resource cannot service them simultaneously. Resource allocation corresponds to coloring the graph to minimize the proportion of adjacent nodes that have the same color.

Consider a graph of *N* nodes, each connected to *K* neighbors via undirected edges, randomly constructed in the following way. Distribute the nodes randomly on a unit square. Then, cycling through the nodes in a fixed order, connect each node to its nearest neighbor on the square until it has *K* neighbors. Once constructed, this structure is static. The resulting graph may not be planar, but violations will be local. Graphs constructed in this way approximate the connectivity of an idealized communications network, in which each node is connected to all nodes within a fixed radius, but with an important difference. If the radius is fixed, the node degree *K* will vary broadly about some mean value. For example, in one trial with 100 nodes in the unit square and a radius of 0.1, the mean *K* is about 3, but the standard deviation of *K* is 1.7. In this study the focus is on how the dynamics of graph color-

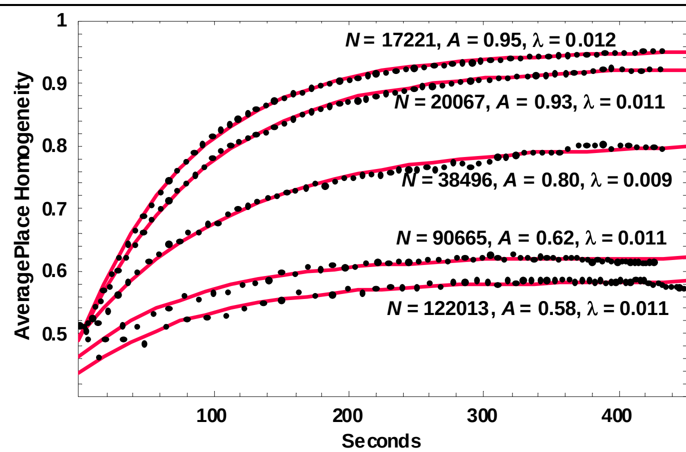

Figure 2. Effect of changing document population (all populations start at ~17221 and end at *N*). λ for all curves is essentially the same.

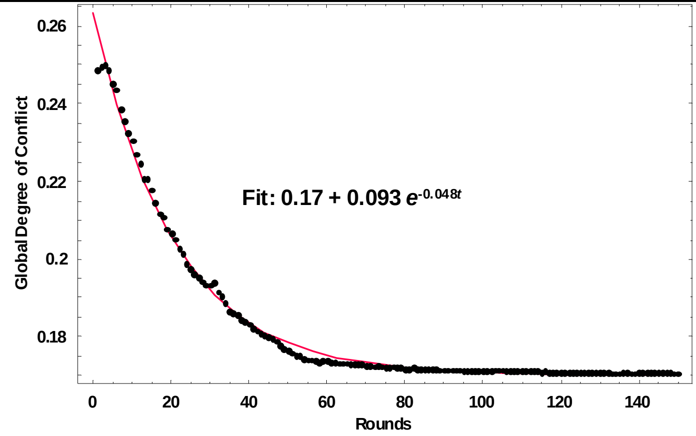

Figure 3. Graph coloring convergence. The solid line is a fit to $A + B \cdot e^{-\lambda t}$.

ing vary with K, so it is better to fix K at a precise value, transferring the variation to the radius at which nodes can be connected.

Agents.—Each node in this graph is an agent, and executes a soft real-time graph coloring algorithm [3, 7]. The agents execute synchronously in discrete time. At any step, each agent has one of G colors. The assignment of colors may change over time. An agent can perceive its neighbors' colors after a delay.

Global Objective.—The objective is to minimize the g*lobal degree of conflict*, computed as the number of edges in the graph that connect two nodes with the same color, times G, divided by the overall number of edges. The global degree of conflict is 0 for a perfectly colored graph, about 1 if all agents make random choices, and greater than 1 if conflicts are worse than under random choice.

Local Decisions.—Agents are selected randomly without replacement in each round. All agents share a global activation level (the probability that the agent will activate at a given round). If activated, an agent re-evaluates its color assignment based on the local Degree of Conflict, (the number of neighbors that share the node's color divided by the overall number of neighbors K). The node calculates the degree of conflict for each of the G possible colors, using the perceived color of its neighbors. It compares these degrees of conflict with that of its current color, and selects the color yielding the lowest local degree of conflict. Because this decision takes into account only the colors of immediate neighbors, it is local. It is also noisy, since the information about neighbors' colors is delayed and may be inaccurate.

Convergence.—Figure 3 shows the convergence of a run (Table 2) and the excellent exponential fit.

2.3 Area Surveillance [12]

Motivation.—A common task for uninhabited robotic vehicles is surveillance of a region of territory. Such surveillance must satisfy several characteristics. One is that vehicles should spread out over the area to avoid double coverage and reduce the time needed to cover the entire area.

Agents.—Each vehicle is an agent in this system.

Global Objective.—A convenient metric for such a system is how rapidly the agents initially cover the territory that they must monitor, tracking the fraction of the area that has been seen as a function of time.

Local Decisions.—A simple algorithm for this problem uses digital pheromones [8, 10, 11]. The environment is represented in the pheromone infrastructure as a square grid, each cell of which has a place agent.

1. Once a second, each place agent deposits twenty units of attractive pheromone in its cell, propagates pheromone to the eight neighboring cells, and evaporates the pheromone by a fixed proportion.
2. Every time a vehicle enters a new cell (on average, once every 4.8 seconds), it deposits two units of repulsive pheromone and zeros out the attractive pheromone in its current cell.
3. Once every twelve seconds, each place evaporates (but does not propagate) its repulsive pheromone by a fixed proportion.
4. Each vehicle moves to the neighboring cell for which difference (attractive pheromone – repulsive pheromone) is greatest;

Agents' decisions use only by the information available in their immediate vicinity, and thus are local (though the propagation of attractive pheromone in step 1 provides some spread of information over time). In the absence of a vehicle, step 1 leads to an asymptotically constant level of attractive pheromone in each cell, drawing in vehicles. Step 2 causes the vehicles to spread out from one another, and avoids repeat visits that are close to one another. Because step 1 repeats after step 2, and because the repulsive pheromone from step 2 evaporates over time, eventually each site will be revisited.

Convergence.—Table 3 shows the parameters for a run of this system. For the fastest possible coverage, at each time step, each vehicle should move immediately to a cell that has not yet been visited. Such a strategy is physically impossible, because it would require vehicles to move directly between noncontiguous cells. But it provides an upper limit, visiting 15 new cells or 15/40000 = 0.0375% on each time step. Figure 4 shows both this upper limit and the actual convergence of the algorithm, in steps of 4.8 seconds (the average time it takes a vehicle to move from one cell to another). The convergence is *not* exponential, but linear with a slope of 0.030.

Table 3. Parameters for Area Surveillance

Size of Area (cells)	200x200
Number of Vehicles	15
Attractive pheromone	
Update interval	1 sec
Deposit size	20
Evaporation rate	0.03
Propagation rate	0.75
Repulsive pheromone	
Update rate	~4.8 sec
Deposit size	2
Evaporation rate	0.3
Propagation rate	0

Table 2. Parameters for Graph Coloring

Activation Level		2.5%
Communication Latency		1 round
Number of nodes N		600
Node degree K		150
Number of colors G		4

3. THEORY

To understand these behaviors, we study a simple model that retains many of the features common to these applications, an extension of the adaptive walk [6]. This model converges exponentially fast, and the parameters that govern this convergence provide useful guidance in designing and analyzing examples such as those we have just discussed.

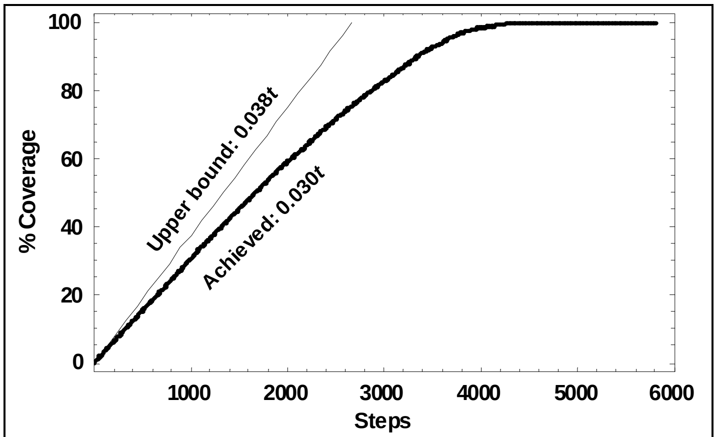

Figure 4. Area surveillance convergence: upper bound and achieved.

3.1 The Adaptive Walk Model

Consider a binary vector $S \in \{0,1\}^N$ of length N. Initially, all elements of S are 0. At each time step,

1. Select an element of S at random.
2. If the element is currently 0, set it to 1 with probability p_{01}. If it is currently 1, set it to 0 with probability p_{10}. Note that p_{01} and p_{10} are independent. In particular, there is no requirement that they sum to 0.

The objective of this system is to maximize $N_1 = \Sigma S$, the number of elements of S that are set to 1.

Though simple, this model has the essential features shared by our more realistic systems. In Section 4, we will map each example onto the adaptive walk. For now, the following general correspondences will motivate the argument.

- Each element of S is an agent, and the array corresponds to the entire system of agents.
- The system objective is global over the entire system.
- The agents do not have access to this global measure in making their decisions. In fact, in this simple model, they do not consider the state of *any* other agent in making their decisions, but choose probabilistically. p_{01} reflects the probability that their local decision will advance the global goal, while p_{10} reflects the likelihood that their local decision will oppose the overall system objectives.

3.2 Analysis

Our extended adaptive walk can be modeled as is a finite state Markov chain, which is known to converge exponentially [4]. We derive the analytic form of this convergence from a mean field analysis in the continuous limit to facilitate subsequent analysis of our example problems.

Consider the time rate of change of $N_0 = N - N_1$ (the number of 0's in S). The analysis for N_1 is similar.

On a given step, the algorithm will select a 0 element with probability N_0/N and a 1 element with probability $(N - N_0)/N$. If it selects a 0 element, it will turn it to a 1 with probability p_{01}, decrementing N_0 on average by p_{01}. Thus the effect on N_0 due to changes of 0s to 1's is $p_{01}\, N_0/N$. Similarly, if the algorithm happens to select a 1, it will turn it to 0 with probability p_{10}, thus incrementing N_0 on average by p_{10}, with net effect $p_{10}\,(N - N_0)/N$. The total effect is given by the master equation

$$\dot{N}_0 = p_{10}\frac{(N - N_0)}{N} - p_{01}\frac{N_0}{N} = p_{10} - \frac{(p_{10} + p_{01})}{N} N_0 \qquad \textbf{Equation 1}$$

This simple differential equation is of the form $\dot{x} = A - Bx$ and has the solution $x = A/B + C \cdot e^{-Bt}$, or $N_0 = \frac{N \cdot p_{10}}{(p_{10} + p_{01})} + C \cdot e^{-\frac{(p_{10}+p_{01})}{N}t}$ where C is a constant of integration. At time $t = 0$, $N_0 = N$, requiring $C = \frac{N \cdot p_{01}}{(p_{10} + p_{01})}$. To simplify the notation, we write

$$\lambda \equiv (p_{10} + p_{01})/N \ , \qquad \textbf{Equation 2}$$

yielding

$$N_0 = \left(p_{10} + p_{01} e^{-\lambda t}\right)/\lambda \ . \qquad \textbf{Equation 3}$$

Similar reasoning for N_1 yields

$$N_1 = p_{01}\left(1 - e^{-\lambda t}\right)/\lambda \qquad \textbf{Equation 4}$$

3.3 Discussion

The dynamics of both N_0 and N_1 are governed by $\lambda \equiv (p_{10} + p_{01})/N$, which determines both the asymptotes and the speed of convergence of the system.

The speed of convergence increases with both p_{01} and p_{10}. Low values of either mean that the system is likely not to change its state on a given step, so more steps will be necessary to converge. The speed of convergence decreases with N. Each step changes only one element of S, and the influence of a change to one element decreases with the total number of elements. If time is measured in units of N steps, giving each element on average one chance to execute a decision cycle, this factor disappears.

Equation 3 shows that N_0 decreases exponentially from an initial value of N to a final value of $\frac{p_{10}}{\lambda} = \frac{p_{10} \cdot N}{(p_{10} + p_{01})}$. It is interesting that erroneous local decisions do not slow convergence, but rather increase the deviation from optimal that the system is able to achieve. The amount of this deviation increases with N and decreases with p_{01}.

Similarly, Equation 4 shows that N_1 increases exponentially from an initial value of 0 to a final value of $\frac{p_{01}}{\lambda} = \frac{p_{01} \cdot N}{(p_{10} + p_{01})}$. The higher p_{01}, the closer to the optimum the system can come.

Although our analysis is based on the continuous limit, it is remarkably accurate for reasonably sized populations of agents. Figure 5 plots the average results across 20 runs of the adaptive walk with $N = 100$, $p_{01} = 0.9$, and $p_{10} = 0.1$, along with the theoretical model (solid line) and the actual fit (dashed line, obscured by the data points). The difference between the parameters of the theoretical equation and those of a least-squares fit to the data is less then 5%.

4. ANALYZING THE EXAMPLES

The model is close enough to yield useful insights when applied to our three examples. Throughout this section, “the model” is the adaptive walk, and “the application” is the specific example.

4.1 Ant Clustering

In ant clustering, the content agents correspond to the elements of S. Since the global objective is the average homogeneity of a single place, we focus on the convergence of a single place. The model’s dynamic of choosing an element of S and changing its state corresponds to two events in ant clustering: the departure of one content agent that judges the place less similar to itself than its destination, and the entry of another content agent that judges the place more similar than its original place. There is no guarantee that every content agent that leaves a place will be replaced by an incoming one, and in practice some places grow in size while others shrink. But the approximation of constant place population is in the spirit of our simplified model. In the light of this mapping between the model and the application, we compare the shape, speed, and limits of the system’s convergence between the model and the experimentally observed result $0.95 - 0.46e^{-0.012t}$.

Shape.—One might question whether the exponential convergence of the model’s global objective (number, or more usefully percentage, of 1’s) will be preserved in the objective of the application (place homogeneity). To map one to the other, we compute the average pairwise homogeneity among elements in S, under the assumption that two elements of the same value have a similarity of 1 while two elements of opposite value have a similarity of 0.

First, construct all possible pairs of elements, sum their similarities, and divide by the total number of pairs. The total number of pairs is just the binomial coefficient $\binom{N}{2}$. The only pairs that contribute to the total pairwise similarity are those between elements of the same value; there are $\binom{N_1}{2}$ pairs of 1’s and $\binom{N-N_1}{2}$ pairs of 0’s. Thus the average pairwise similarity is $\dfrac{\binom{N_1}{2}+\binom{N-N_1}{2}}{\binom{N}{2}} = \dfrac{2N_1^2 + 2N_1(N-N_1) + N(N-1)}{N(N-1)}$.

To simplify further, consider the limit of large N, where $N \sim (N - 1)$, and let $r \equiv N_1 / N$. Then the homogeneity of S becomes

$$2r^2 + 2r - 1 = 0.5 + 2(r - 0.5)^2 . \qquad \textbf{Equation 5}$$

Place homogeneity decreases with increasing r for $r < 0.5$, then increases. In ant clustering, with cluster agents initially assigned randomly to places, r begins at 0.5, so it is natural to see the factor $(r - 0.5)$ as reflecting the degree of convergence of an adaptive walk that begins with a random S rather than one that is uniformly 0. Thus, to a first approximation, place homogeneity varies as the square of r, which is an exponential with rate 2λ. The theory thus explains the exponential shape of the experimental results.

Asymptote.—The model predicts $r = \dfrac{N_1}{N} = \dfrac{p_{01}}{(p_{01}+p_{10})}\left(1-e^{-\lambda t}\right)$, with asymptote $p_{01}/(p_{01}+p_{10})$. Translation to homogeneity (Equation 5) yields

$$A = \frac{p_{01}{}^2 + p_{10}{}^2}{(p_{01}+p_{10})^2} \qquad \textbf{Equation 6}$$

Experimentally, A = 0.95, suggesting $p_{10} = 0.026 p_{01}$. (The simplicity of the model does not warrant such precision, but we carry out the computation for the sake of illustration.) Thus our algorithm is less than 3% as likely to take a backward step as a forward one. The uncertainty in our local decision is due entirely to partial sampling, not to any noise in the data, so it is reasonable that backward steps should be rare. The effectiveness of forward steps can be estimated by analyzing the speed of convergence.

Speed.—The experimental results in Section 2.1 covered two cases: clustering of a static document population, and clustering when the population grew at a constant rate. Our model supports a quantitative analysis of the first case, and (at present) a qualitative analysis of the second.

First, consider the convergence with a fixed population. We are concerned with the average homogeneity of a single place, of average population 17221/900 ~ 20, so we expect r to grow as $\lambda = (p_{10} + p_{01})/20$, and place homogeneity as $2\lambda = (p_{10} + p_{01})/10$. We do not know p_{01} and p_{10}, but from the asymptote we know that when $p_{01} = 1$ (its maximum), $p_{10} = 0.053$, so an upper bound on 2λ would be 0.1053.

The application converges as $e^{-0.012t}$, where t is measured in seconds. On average, the application activates individual clustering agents 558 times each second, and these agents are distributed across 900 places. So on average an agent in the place we are considering is activated once every 558/900 ~ 0.6 seconds. The move of such an agent out of the place is only half of an update step. The other is the move of an agent from another place into our designated place. Other agents have 900 places from which to choose, and only one of these will lead to our place, which will happen once every 0.6 seconds. So the effective time required for the system to make one step is 1.2 seconds. Let q be the number of such steps. Then $t = 1.2q$, and the observed convergence rate $0.012t = 0.014q$, suggesting that $2\lambda \sim 0.014$. This is within the upper bound suggested by the theory, and not so low as to be unreasonable. It would be exact if $p_{01} \sim 0.136$ and $p_{10} \sim 0.004$ At first glance, we would be disappointed to think that a content agent’s action advances the global objective so infrequently. More likely, the discrepancy is due to the incremental nature of the improvement achieved by replacing one document with another that is more similar to the place’s population. This improvement is not

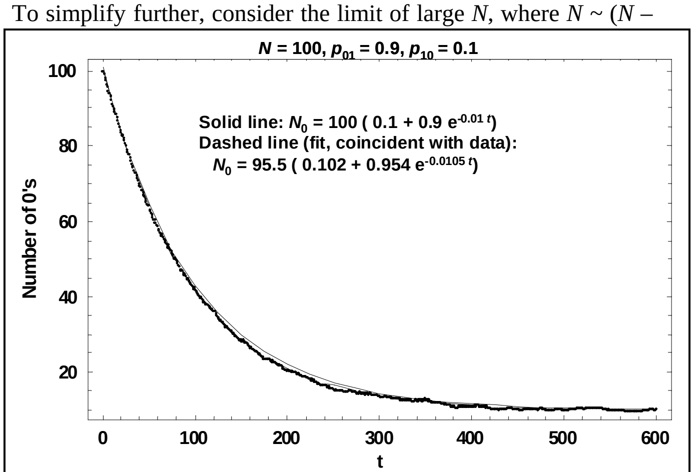

Figure 5. Theoretical vs. actual convergence of adaptive walk.

a binary change as in the adaptive walk, but a more incremental shift, analogous to incrementing an element of S by a real in [0,1]. If we think of the p_{01} parameter as the product of the probability of a beneficial move times the amount of benefit conferred, the estimated value of 0.13 might reflect (for example) a 65% probability of making an improving move times a 20% improvement.

We do not yet have a quantitative analysis of the growing document population, but the model offers qualitative insight into the remarkable stability of the exponent λ as the growth rate increases. Equation 2 supports our intuition that higher N should yield lower λ, all other things being equal. But other things are not equal. The random assignment of new documents to places reduces the accuracy of a document's decisions about how consistent it is with its own place and with alternatives. This uncertainty should increase the probability of a backward step p_{10}, increasing the numerator of Equation 2. In other words, the higher rate of backward steps counterbalances the retarding effect of increased population. An increase in p_{10} relative to p_{01} as the population increases is consistent with the decrease of the asymptote, according to Equation 6. Naïve application of Equation 6 to an asymptote of 0.6 yields $p_{10} = 2.6 p_{01}$, corresponding to an increase in $p_{10} + p_{01}$ by a factor of more than three that will counterbalance the increase in N in the numerator of λ. We should not be surprised that the correction is not exact numerically, since our theory does not yet provide for changes in the length of S during the adaptive walk. However, the static results do suggest qualitatively how λ might plausibly remain constant as N increases.

4.2 Graph Coloring

One might be tempted to map each agent of the graph coloring problem (the nodes) onto the elements of S, just as we did for the ant clustering agents. This will not work. The basic dynamic of the model maximizes ΣS, and the counterparts to the elements of S must be stateful entities such that we want to maximize the number of entities in a given state. The basic state of each node in the graph is its color. The global objective, minimal global degree of conflict, is achieved when the nodes have the most diverse range of colors possible, not when they are all colored the same.

A more appropriate counterpart for elements of S is found in the edges of the graph. The state of an edge is 0 if it joins to nodes of the same color, and 1 otherwise. A monotonic decrease in the number of 0's corresponds to a monotonic decrease in the global degree of conflict. This mapping is instructive because the elements in the adaptive walk do not correspond to the decision-making agents in the application. Each node decision will affect K links, and we shall have to take this into account in our analysis of the speed of convergence. Let us compare the theoretical predictions for this system with the observed experimental fit, $0.17 + 0.093e^{-0.048t}$, in shape, speed, and asymptote of convergence.

Shape.—As with ant clustering, we must first consider the relation between the model's objective function ΣS and the application's objective function of global degree of conflict. The latter is number of edges in the graph that connect two nodes with the same color, times the number of colors available, divided by the total number of edges. In the model, the objective function is thus ($2G\Sigma S/nK$), the same as ΣS up to a constant, and the time steps are the same up to a constant (K), so the shapes will be the same.

Asymptote.—In this problem, we are interested in minimizing rather than maximizing the objective function, so we use N_0

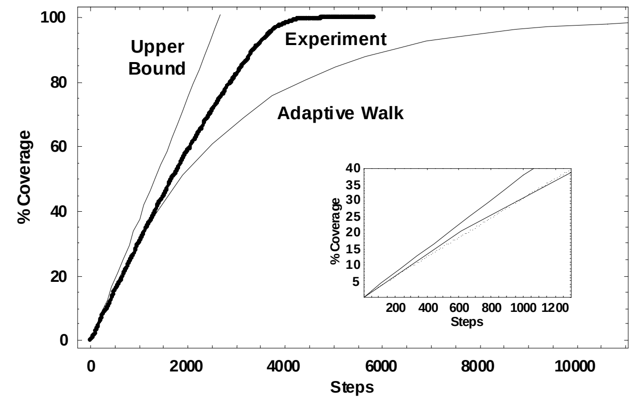

Figure 6. Comparison of adaptive walk with area surveillance.

(Equation 3) rather than N_1 as our model. In terms of Equation 3, our objective function is (recalling that $nK/2 = N$ and $G = 4$)

$$\frac{2GN_0}{nK} = \frac{4N_0}{N} = \frac{4}{(p_{10} + p_{01})}\left(p_{10} + p_{01}e^{-\lambda t}\right)$$

and the asymptote is $4p_{10}/(p_{10} + p_{01})$.

Experimentally, our asymptote is 0.17, suggesting $p_{10} = 0.04\, p_{01}$. This upper bound assumes that the non-zero asymptote is due only to a non-zero p_{10}. In the application, another constraint prevents the system from reaching a 0 asymptote: $K > G$, so it is impossible for even a single node to avoid conflict with all of its neighbors. We can estimate this effect by setting the communication latency to 0. At this setting, the asymptote is still 0.17, suggesting that the nonzero asymptote is due solely to the overconstrained nature of the problem, and $p_{10} = 0$ even with a communication latency of 1. The activation level of 2.5% is apparently low enough to delay the activation of nodes long enough to permit information about their neighbors to reach them.

Speed.—To compute λ and thus p_{01}, we need to align the time steps between the model and the application. Two factors are involved. First, the time axis in Figure 3 is in rounds, each of which updates all n nodes. Second, each node update changes the state of K edges. So one time step in Figure 3 corresponds to nK steps in the adaptive walk. Thus the convergence constant of 0.048 in the application corresponds to $0.048/nK$ for the adaptive walk, and it is this value that should be compared with λ. $N = nK/2$, so $2(p_{10} + p_{01})/nK = 0.048/nK$, and $(p_{10} + p_{01}) = p_{01} = 0.024$.

These figures apply to the model, and thus to link updates. In terms of the nodes, each node update changes K links, yielding $p_{01}K$ improved links out of $nK/2$. So the probability that a node will make a useful change is $2p_{01}/n$, or 8E-5 for our parameters. Thus while the update rule for the distributed graph coloring algorithm always moves in the right direction, its contribution is very small, requiring long times for convergence. This slowness is not surprising, considering the highly over-constrained nature of this system. Again, the extreme simplicity of the model compared with the application urges us to take these results *cum grano salis*.

4.3 Area Surveillance

Our final example does not exhibit exponential convergence. Nevertheless, it is worthwhile to compare the experimental results and theoretical upper bound we have already presented with the adaptive walk. Like the upper bound computation, the adaptive walk is unrealistic, since it does not respect constraints on vehicle movement, but assumes that any vehicle can move immediately to any cell in the area, even one remote from its current location.

At each step in area surveillance, 15 vehicles are sampling cells of a 40000-cell area. In terms of the model, this is comparable to $|S|$ = 40000/15 ~ 2667. For p_{01} = 1, p_{10} = 0, we have λ = 3.75E-4 (the same as the slope of the upper bound discussed in Section 2.3), and $N_1 = 2667\left(1 - e^{-3.75E-4t}\right)$. Figure 6 plots this line together with the upper bound and the observed data.

In spite of its simplicity, the model provides an excellent fit to the experimental data up to about 40% coverage. The inset shows how the model rises slightly faster than the experiment, then begins to fall below it. We attribute the rapid rise to the assumptions of the model: while the air vehicles are constrained by the need to move between contiguous cells, and so must often repeat coverage of cells that have already been seen, the model can go directly to any cell. Of more interest is the shortfall above 40% coverage. The adaptive walk slows as more and more of the area is covered, so that the random selection of the next site to visit frequently selects a site that has already been visited.

The continued straight-line progress of the experiment shows the effectiveness of the pheromone mechanism in improving over the random selection of the next site to visit. This improvement arises because pheromones reduce the locality of the decision process, in two ways. First, the propagation of attractive pheromone makes information from one cell available in a neighboring cell, reducing spatial locality and generating a gradient that guides the movement of vehicles. Second, the persistence of pheromone deposited by one vehicle for sensing by another reduces the temporal locality of decisions, enabling decisions at one point in time to take into account the results of previous decisions.

From an engineering point of view, this example illustrates how the adaptive walk model can provide a lower bound for estimating the achievable performance of a real system, and for measuring the effectiveness of mechanisms for overcoming locality.

5. CONCLUSIONS

Many modern applications require systems that can make decisions based on local knowledge to advance a global objective. Such systems are interesting both theoretically and from an engineering perspective. Theoretically, it is interesting that they often (though not always) exhibit exponential convergence. To deploy them effectively, we need to understand how to predict, control, and evaluate their convergence.

An enhanced version of the adaptive walk is simple enough for theoretical analysis, yet real enough for engineering study. Our experience with this model yields the following insights.

1. Exponential convergence emerges from the simplest stochastic search, and thus should be very common in such systems. This is a fortunate result, since such convergence supports any-time processing in which solution quality increases very rapidly at the outset, yet continues to improve if more time is available.
2. The speed of convergence is a function of three parameters: the number of interacting agents, the probability that an agent whose state needs to change will change (advancing the objective), and the probability that an agent whose state does not need to change will change anyway (hindering the objective).
3. The model indicates two routes to improving the convergence of such systems. First, one can use mechanisms such as pheromones to propagate information through space and time, thus reducing the locality of the decisions. An example is the use of pheromones to guide robots in the area surveillance application. Second, one can focus on the three parameters mentioned in the previous step. An example is the use of pheromone learning [9] to favor the activation of agents most likely to improve the overall system behavior, thus effectively reducing *N* and increasing speed of convergence.
4. The model provides a template for assessing the performance of a real application. If an application exhibits exponential convergence (as in the clustering and graph coloring examples), its asymptote can be used to estimate the relative size of p_{10} and p_{01}, and thus how effectively the system avoids backward steps in favor of forward ones. Then its speed of convergence can be used to estimate the actual value of these parameters. If an application does converge exponentially (as in the area surveillance example), the model provides a baseline that can be used to judge the effectiveness of the application's mechanisms.

Our model is extremely simple, and we must be cautious in applying it to real systems. We have seen two examples of possible confusion in our analyses. 1) In the clustering application, the estimated p_{01} actually includes the *size* as well as the *likelihood* of a desirable step, and similarly, p_{10} will conflate the size and likelihood of an undesirable one. 2) In the graph coloring application, the asymptote depends on structural constraints that are not reflected in the model, and might lead an analyst to overestimate p_{10}. In spite of such limitations, the model provides a hitherto unavailable framework for the disciplined analysis of distributed decentralized systems, and we offer it as the foundation for enhancements that will make it even more useful (such as incorporating dynamic change in the size of the population of elements).

6. ACKNOWLEDGMENT

This study was supported in part by DARPA under contract F30602-99-C-0202. The views and conclusions in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency, the Department of Defense, or the US Government. This study was supported and monitored in part by the Advanced Research and Development Activity (ARDA) and the National Geospatial-intelligence Agency (NGA) under contract number NMA401-02-C-0020. The views, opinions, and findings contained in this report are those of the author(s) and should not be construed as an official Department of Defense position, policy, or decision, unless so designated by other official documentation.

7. REFERENCES

[1] Brueckner, S. and Parunak, H. V. D. Resource-Aware Exploration of Emergent Dynamics of Simulated Systems. In *Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS 2003)*, 2003, 781-788.

[2] Deneubourg, J. L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C., and Chretien, L. The Dynamics of Collective Sorting: Robot-Like Ants and Ant-Like Robots. In J. A. Meyer and S. W. Wilson, Editors, *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior*, MIT Press, Cambridge, MA, 1991, 356-365.

[3] Fitzpatrick, S. and Meertens, L. Soft, Real-Time, Distributed Graph Coloring using Decentralized, Synchronous, Stochastic, Iterative-Repair, Anytime Algorithms: A Framework. KES.U.01.5., Kestrel Institute, 2001.

[4] Goel, S. Three Approaches to Finite Markov Chains. Cornell University, Ithica, NY, 2004. http://www.cam.cornell.edu/~sharad/papers/FMC.pdf.

[5] Jain, A. K., Murty, M. N., and Flynn, P. J. Data Clustering: A Review. *ACM Computing Surveys*, *31*, 3: 1999.

[6] Kauffman, S. A. and Levin., S. Toward a general theory of adaptive walks on rugged landscapes. *J. Theoret. Biol.*, *128*: 1987, 11-45.

[7] Meertens, L. and Fitzpatrick, S. Peer-to-Peer Coordination of Autonomous Sensors in High-Latency Networks using Distributed Scheduling and Data Fusion. KES.U.01.09, Kestrel Institute, 2001.

[8] Parunak, H. V. D., Brueckner, S., and Sauter, J. Digital Pheromones for Coordination of Unmanned Vehicles. In *Proceedings of Workshop on Environments for Multi-Agent Systems (E4MAS 2004)*, Springer, 2004, (forthcoming).

[9] Parunak, H. V. D., Brueckner, S. A., Matthews, R., and Sauter, J. Pheromone Learning for Self-Organizing Agents. *IEEE SMC*, *35*, 3 (May): 2005, 316-326.

[10] Parunak, H. V. D., Brueckner, S. A., and Sauter, J. Digital Pheromone Mechanisms for Coordination of Unmanned Vehicles. In *Proceedings of First International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2002)*, 2002, 449-450.

[11] Parunak, H. V. D., Purcell, M., and O'Connell, R. Digital Pheromones for Autonomous Coordination of Swarming UAV's. In *Proceedings of First AIAA Unmanned Aerospace Vehicles, Systems,Technologies, and Operations Conference*, AIAA, 2002.

[12] Sauter, J. A., Matthews, R., Parunak, H. V. D., and Brueckner, S. A. Performance of Digital Pheromones for Swarming Vehicle Control. In *Proceedings of Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2005, (forthcoming).

[13] Weinstein, P., Parunak, H. V. D., Chiusano, P., and Brueckner, S. Agents Swarming in Semantic Spaces to Corroborate Hypotheses. In *Proceedings of AAMAS 2004*, 2004, 1488-1489.