Simulation of Swarm Intelligence-Based Message Broadcast in Highly Mobile Ad Hoc Networks

Michael Kirkpatrick[*]
Ryan Kennedy
Gregory Madey
University of Notre Dame
Notre Dame, IN 46556
gmadey@nd.edu

Keywords
Ad Hoc, Mobile, Network, Swarm

ABSTRACT

This paper presents a means of simulating message passing in highly mobile multi-hop ad hoc networks and uses it to explore the potential for the efficient broadcasting of messages by treating the nodes in the network as autonomous agents for which only a simple set of rules governing the passing of messages from one node to another is defined.

The general problem presented by highly mobile systems is one of coordinating communication between nodes in settings in which the details of their connections to each other—the portions of the network that are within radio range at any given time—can change. Several scenarios in which a swarm intelligence-based paradigm is applied to this problem are simulated and the results are compared in order to analyze the performance of various approaches relative to each other. A baseline is established in which agents in the system can potentially move after each transmittal of a message to their immediate neighbors. This is then used to evaluate the relative gains in performance achieved by fully exploiting the ad hoc networks as they emerge. The effects of the shape of the deployment area on communication efficiency are also examined. A "biologically inspired" heuristic for broadcasting is introduced. This heuristic has low computational overhead making it especially suitable for "power-limited" systems.

INTRODUCTION

With the proliferation of portable computing and communication has come an increasing need for communication networks that can support devices that move around as they are working. This has been addressed in part by radio based local networks within buildings and by stationary antennas in telephone cells. Each of these types of methods, however, relies on a portion of the network being fixed in place. For example, the "on air" network hub connecting laptops moving within an office is installed in a fixed location, both physically and in terms of its logical connection to the larger network.

In such a setting a computer wishing to communicate with another must pass a message to the hub which may then send it directly to the intended destination or may route it through other parts of the stationary network. This will be done even if the sending and receiving computers are physically within inches of each other. Work has been done to develop communication protocols that exploit the fact that computers in radio range of a central hub may also be in radio range of each other and to allow direct communication to occur in such cases [1, 2].

In some situations, such as when all of the nodes of the network are mobile, it may even be desirable to eliminate the reliance on fixed network elements entirely. Examples of such networks include robots exploring the surface of a distant planet, soldiers moving on a battlefield, robots searching for avalanche victims [3], or the elimination of fixed towers in cellular telephone systems.

The general problem presented by these mobile systems is that the details of their connections to each other can change. Inouye, et al. [2], describe algorithms that test for connectedness and route messages accordingly, adapting to changes in the configuration of the system. A common feature of such algorithms is an attempt to establish and maintain some type of global knowledge about the current connections either within the system as a whole or on a more localized sub-system, called an *ad hoc network*.

Nature provides us with examples of mobile, independently operating agents that seemingly work together to perform tasks in a highly efficient manner without complex communication networks and without global knowledge of the locations of individuals. One such example can be found in flocks of birds flying in formation. No individual bird is aware of the positions of all of the other birds. No specific bird directs the movement of the flock. Instead, each bird takes its cue to turn in one direction or another from those immediately surrounding it. Yet, when viewed as a whole the flock appears to be moving in an intelligent, directed manner [4]. This method of groups performing tasks effectively by using only a small

---

[*] Michael Kirkpatrick is currently at West Virginia Univ.

set of rules for individual behavior – called *swarm intelligence*—has shown promise when applied to routing problems such as those encountered in transmitting messages through telephone and computer networks [5].

The topology of such networks is generally fixed. Changes may result from equipment failures or from the addition or removal of nodes but these events are relatively infrequent. In this paper we apply this swarm intelligence based approach to networks that, by nature, have frequent topology changes. We are interested especially in developing an effective means of simulating these networks and then using the simulations to study message passing within them.

In doing this we explore the potential for providing reasonably efficient and reliable communication between nodes in an entirely mobile network – one without any necessarily stationary parts—by defining only a simple set of rules governing the passing of messages from one node to another. In such a network, messages may be passed between two nodes within radio range of each other. A message intended for a third node not in range of either of them may be passed from the first node to the second and held by the second until it either encounters the intended recipient or is able to pass it on to yet another node to carry.

As a starting point for studying this problem, a simulation was developed which models a number of agents moving independently and at random within a defined space. When two agents encounter (come within range) of each other, if one of them has a message and the other does not, the message is passed between them so that both of them have it. So two simple rules govern the entire operation of the agents:

1. Move according to a (possibly random) schedule independent of other agents;
2. When encountering another agent, pass a message to or receive a message from that agent, as appropriate.

A third rule, which determines whether an agent retains a message indefinitely, "times it out," replaces it with a newer message, or drops it upon an acknowledgement of receipt, can also be defined to control the retention of a message by an agent.

LITERATURE REVIEW

Networks in a mobile environment are characterized primarily by the fact that the connections between nodes can change. A common view of such a network is of nodes which are connected to each other and/or to some stationary network by a wireless medium such as radio [6, 7]. At any point in time a given node will be within range of some subset of the nodes in the network. The number of nodes in this subset can range from zero (in which case the node is isolated) to $n-1$, where n is the total number of nodes in the system. Viewing this connectivity from the system's perspective, rather than from that of a node, gives a picture of a partially connected graph made up of connected subgraphs, or groups [7]. These subgraphs, referred to in the literature as ad hoc networks, allow communication to take place for a while but do not guarantee the existence of a link at arbitrarily selected times.

Most communication in these networks tends to be either broadcast (flooding) or point-to-point [6]. Perkins and Bhagwat [8] describe a method in which each node maintains a routing table of the nodes it can reach and periodically broadcasts this table (or otherwise makes it available) to its neighbors. Johansson, et al. [9], compare the efficiency of maintaining such tables proactively, as done by Perkins and Bhagwat, to building them as needed and, in so doing, saving the overhead of maintaining tables with links to nodes which are no longer in communication. They also use the past history of routes in the ad hoc network to make predictions of the stability of paths and attempt to use more stable paths. Chandy and Lamport [10] describe a method of taking "snapshots" of a distributed system showing its state at a given point in time and Murphy, et al. [11], adapt this technique for use in delivering messages in an ad hoc network.

A good summary of issues relating to data communications in a mobile setting is given by Perkins [12]. Of particular interest, he says that the overall bandwidth can be improved by increasing the number of base stations in a system which includes stationary base stations. Adler and Scheideler [6] consider power-controlled ad-hoc networks. In such a system, the ability of a given node to connect with another is not entirely a function of the chance that the nodes are positioned in such a way that they can communicate (either directly or through other nodes in the ad hoc network). Instead, a node may be able to boost or reduce its signal, allowing it to communicate farther or to restrict the recipients.

Work has also been done studying the passing of messages between vehicles traveling on highways [13, 14, 15], allowing a braking car, for example, to notify the cars behind it that it was slowing, giving more reaction time than that provided by the driver simply seeing the brake lights of the car immediately ahead.

One of the more significant problems common to message delivery in mobile settings, in general, and in highly mobile settings, in particular, is the overhead of maintaining routing information. The nature of highly mobile environments is such that this information is often not available "naturally." Das, et al. [16] offer an algorithm for multicasting in a highly mobile network which reduces overhead and increases multicast efficiency by assigning specific forwarding tasks to certain nodes and Chen and Liestman [17] explore an approach which uses weakly connected dominating sets to cluster nodes, simplifying the ad hoc networks.

Williams and Camp [18] provide a good summary of twelve broadcast protocols with comparisons between them. And Peng and Lu [19] describe ways of reducing redundancy of message transmissions in mobile ad hoc networks. They suggest, for example, to delay relaying a message until an agent has had a chance to move out of the area in which it received the message and Hass, et al. [20] suggest a probability-based approach to determining whether or not to relay a message when given an opportunity to do so.

Finally, Obraczka, et al. [21, 22] discuss a special case in which the agents are very highly mobile, or “fast moving.” This last scenario—and the broadcasting of messages through a process of “flooding”—is most closely related to the work we report in this paper.

SIMULATION ENVIRONMENT

For this research, mobile network nodes were modeled as agents in a Swarm [23, 24] simulation. The simulator used was written in Java using the Java Swarm Libraries, open source software tool developed at the Santa Fe Institute to provide support for agent based model simulations. Agents operate according to relatively simple sets of rules and do so with little or no global knowledge of the system. The rules defining behavior instruct each agent how and when to move and how to interact with other agents. Global knowledge, if any, is limited to static information such as the total number of agents in the system, the dimensions of the environment, and so on. Dynamically, a given agent is aware of only its own state and its immediate surroundings.

Of primary importance to the concept of agent based modeling is the idea that the agents are autonomous. According to Chantemargue, et al. [25], autonomy is believed to guarantee and to be a necessary condition for adaptability and self-organization. We are interested here in defining an environment that enables us to observe the emergence of self-organization to facilitate message routing in lieu of a central controller.

Agents can be modeled to have a variety of attributes. Several are listed by Thangiah, et al [26]. Those that are relevant to the work described in this paper are

- Can communicate with other agents;
- Can sense and react to changes in the environment;
- Are capable of long periods of unattended operation;
- There is no central authority governing an agent’s behavior.

Key among these for our purposes is the last point: we are especially interested in the agents’ communication capabilities without reliance on central or regional control.

For this work the environment is simulated as a rectangular (usually square) grid of cells. A cell is either empty or occupied by an agent and, overall, the grid is populated by placing agents in some given percentage of the total number of grid locations. This percentage is called the *density* of agents in the grid. The dimensions of the grid and the density of agents are selected at the beginning of each run. An agent is selected to be the originator of the message. In some runs a destination agent is selected as well. In others the message, is sent to all agents. As the simulation runs, individual agents move randomly, in any of eight possible directions (corresponding to compass points N, NE, E, etc.). Only one agent can occupy a given grid location at a time so if moving in the selected direction would result in a collision, the agent simply sits still for one time step and attempts to move again (in a randomly selected direction) at the next step.

The simulations run can be categorized into four types. These types are described by the following cases:

- Case 1: Messages are passed only to neighbors (that is, agents in any of the eight locations surrounding the agent carrying the message) at each time step;
- Case 2: Messages are passed throughout the ad hoc networks at each step;
- Case 3: Messages are passed as in Case 1 but some agents move in a fixed direction, reversing at the edges;
- Case 4: Messages are passed as in Case 1 but are purged from the agent’s memory after some period of time and an agent, once having carried the message, will not accept it a second time.

Except for the fixed direction agents in Case 3, the agents move randomly at each step and pass messages to other agents with which they have contact—either direct or networked, depending on the case. (In the simulation “direct contact” means physically touching on a side or diagonal.) Initial parameters can be set to vary the frequency at which agents accept messages and the number of time steps an agent will retain a message.

Figure 1. Animation of agents’ motion and message propagation; gray cells represent mobile agents which have received message; white cells represent agents which have not.

Figure 1 shows an animation representing the movement of agents and the passing of a message between them. Figure 2 shows a graph of the message saturation throughout the run.

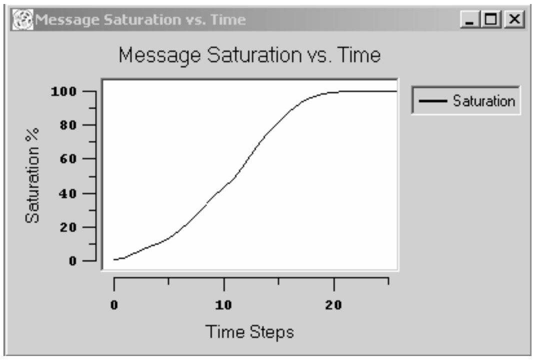

Figure 2. Run-time graph showing message saturation at end of run.

SIMULATIONS

This work focuses primarily on the problem of broadcasting messages in a highly mobile ad hoc network. As noted in Section 2, above, Obraczka, et al. [21,22] make a further distinction of networks in which the agents (nodes) are "fast-moving." The simulation results reported here are expressed in terms of "time steps" which represent the time required for an agent to move from one location to another. It is assumed that communication can take place between two agents within range of each other in less than a single time step.

Several simulations were run, as described in section 3 above, and those results are given here. An analysis is provided in the next section. These runs were made using a 50 x 50 grid for a total of 2500 locations and these locations were populated at random to a given density. Several runs were made for each density setting, using different starting points. In each run the number of time steps required for the message to reach all of the agents was recorded. Also recorded were the number of steps needed for the message to reach 90% of the agents and the average number of steps for the message to reach another agent. This last measurement was made by counting the number of steps required for the message to be passed from the selected origin to every other agent. The average of these step counts was then calculated. The results of the runs for each density were then averaged (mean) over ten runs and are summarized in the tables that follow, starting with Case 1 data shown in Table 1 and graphed in Figure 3.

When graphed as in Figure 3, it is apparent that the number of time steps required to pass the message drops dramatically as the density is increased from, say, 10% to 20% or 30% but changes much more slowly after that.

Table 1. (Case 1) Message passed only to neighbors at each time step

Density	All	Avg	90%
10%	109	51	77
20%	55	23	37
30%	29	12	22
40%	23	9	16
50%	17	7	12

In any given time step, an agent will be connected to (within range of) zero or more other agents. Those agents, in turn, may be connected to still other agents, and so on. This "collection" of connected agents comprises an ad hoc network within the system. Case 2 exploits the existence of these networks by passing messages throughout the networks before advancing to the next time step. This is justified by the realization that radio communication, for example, can take place much more quickly than the agents themselves move. As before, a 50 x 50 grid was used and the density was varied from 10% to 50%. The results of these runs are summarized in Table 2 and Figure 4.

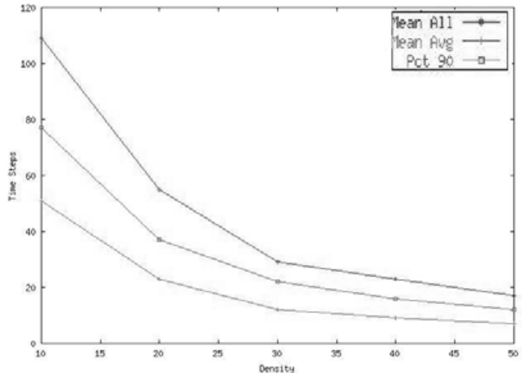

Figure 3. (Case 1) Message passed only to neighbors at each time step

In Case 3 simulations, most of the agents moved as in Cases 1 and 2 but some percent of them was selected to move back and forth across the grid in an otherwise fixed direction. These runs are summarized in Table 3.

In Case 4 simulations the agents held a message for only a given number of time steps after acquiring it and then purged it from memory. In this case, agents that had once carried the message refused to accept it after that from other agents. Here we were interested in the percentage of the agents who received the message before it was lost in the system. Results from these runs are summarized in Table 4.

Table 2. (Case 2) Message passed throughout existing networks at each time step

Density	All	Avg	90%
10%	88	40	65
20%	35	16	27
30%	16	6	11
40%	6	1	1
50%	5	1	1

If the density was 20% or more, all—or nearly all—of the agents received the message before it was lost. In fact, if the density was as low as around 14%, about 90% of the agents would receive the message. Other simulations showed that around 70% of the agents would still get the message when the retention time was cut to five steps. The reason for this can be seen from watching the display during the runs. As shown in Figure 5, when the density is sufficiently high, the message carrying agents form an advancing wall ahead of the area in which agents have had but lost the message.

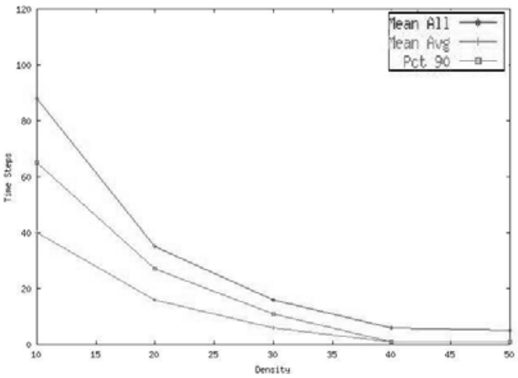

Figure 4. (Case 2) Message passed throughout existing networks at each time step

Table 3. (Case 3) Some agents moving in a fixed direction

Density	Fixed	All	Avg	90%
10%	**10%**	109	47	75
	20%	94	39	61
	30%	108	46	74
20%	**10%**	52	25	39
	20%	57	24	39
	30%	63	24	38
30%	**10%**	31	12	21
	20%	31	12	19
	30%	32	14	23

Table 4. (Case 4) Message retained for 10 time steps

Density	%Delivered
10%	27
15%	96
20%	100

As stated above these simulations all involved a test region defined by a 50 x 50 grid. But the question is raised of whether or not the shape of the region can affect the results. For example, would the number of steps needed to saturate two regions having the same area but different dimensions be approximately the same or would they differ significantly? If the latter is the case, what is the correlation between the dimensions and the required number of steps? This question was addressed for Case 2 in the next set of simulation runs.

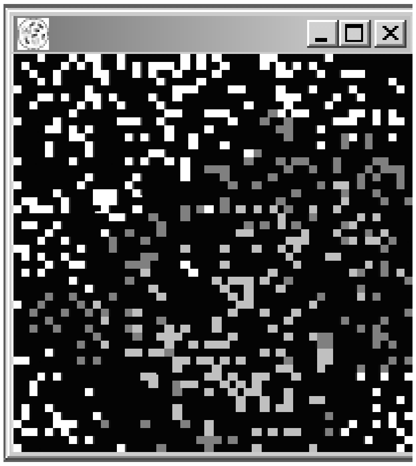

Figure 5. (Case 4) Dark gray agents form "advancing wall" across grid

These simulations compared a number of regions having areas equal, or nearly equal, to 2500. Starting with a long narrow region, the ratio of width to length was increased until the region was a square. The shapes used are described in Table 5.

Table 5. (Case 2) Different dimensions tested

Length	Width	W / L
250	10	0.04
167	15	0.09
125	20	0.16
100	25	0.25
81	31	0.38
61	41	0.67
50	50	1.00

Simulations were run for densities of 10, 20, 30, and 40 percent for each of the regions listed. The number of time steps taken by a message originating near the center of the region to saturate the area was recorded for each run. The results of these runs are summarized in Figure 6.

The dimensions of the regions reported in Figure 6 are given in terms of the ratio of width to length as shown in Table 5. Since the number of time steps required to saturate an area can differ by an order or two of magnitude, depending on the agent density, the step counts for each are normalized to the interval [0.0, 1.0]. These counts for each density are shown plotted for each shape.

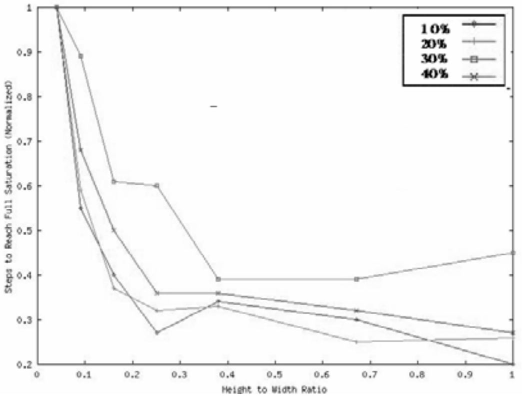

Figure 6. (Case 2) Time steps (normalized) to saturate regions of different dimensions.

In order to examine this method of message passing further and to validate the results obtained, a second simulation was built using StarLogo—an agent-based modeling tool developed at MIT and designed especially for studying the local interactions of individuals in systems exhibiting self-organizing behavior [4, 27].

Table 6. (Case 4 replicated in StarLogo) Saturation percentage for varying density/ retention settings

	Time Steps Retained			
Density	**10**	**15**	**20**	**25**
3%	2	5	5	8
5%	5	10	20	19
8%	---	---	58	86
10%	24	60	93	98
13%	---	---	99	100
15%	82	97	99	100
18%	---	99	---	---
20%	98	100	100	100
25%	100	100	100	100
30%	100	100	100	100

The Swarm and StarLogo simulations, although developed on different tools by different programmers, yield almost identical results (compare Table 4 with the first column of Table 6), thus increasing our confidence in the simulations. Additionally, more runs were made comparing the effects of varying the densities and retention times for Case 4 simulations. These results are summarized in Table 6 and graphed, along with data from runs with shorter and longer message retention times in Figure 7.

ANAYLSIS OF RESULTS

There were four categories of simulations, described above as Cases 1–4, run for this study. In some of them a message, once acquired by an agent, was held indefinitely by that agent. In others, the agent purged the message from its memory after some fixed number of time steps. In most of the simulations the agents moved in a random direction at each step but in one set some percentage of the agents was selected to move in a fixed direction. Of primary interest was the speed with which a message can be dispersed throughout the agents in the modeled "world."

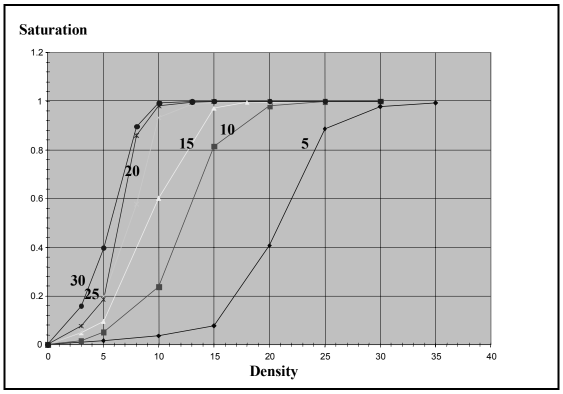

Figure 7. (Case 4) Saturation at different densities for retention times of 5, 10, 15, 20, 25, and 30 steps.

In a "real world" application, the speed of messages between agents or throughout an ad hoc network is likely to be very much greater than the speed at which the agents themselves travel. This means that the time required to pass a message from one agent to another is dominated by the time spent physically moving. Accordingly, the simulation results are expressed in terms of time steps.

Recall that agents have two defining characteristics: they are autonomous and they have no global knowledge of the state of the system. Because of this it is difficult to compare results from these simulations directly to those describing more "typical" networking environments in which the nodes connected at any given time may be known. For this reason, an initial set of simulations (Case 1) was run in which messages were simply passed from an agent to its neighbors at each time step in order to provide a basis for comparison of subsequent results. However, some interesting observations can be made about the data itself.

The measurements taken for this case were the time required (in time steps) for a message to reach all of the agents on the grid (but note that the autonomy of agents and the absence of centralized knowledge of the system's state make it impossible to guarantee delivery of a message to all agents); the time needed to reach ninety percent of the agents; and the average time to reach all of the agents. As can be seen in the run-time graph from a typical run with 20% density, shown in Figure 8, most of the agents receive the message noticeably sooner than the last one receives it. As a result, the average time for all agents to receive the message is only about 40% of that for all agents. Likewise, 90% receive the message in about 70% of the time required for full saturation.

Often the time taken to pass a message from one node in a network to another is reported in terms of "hops," which refers to the number of times the message is passed to an intermediate node. The notion of time steps as used here is similar to that of hops except that time steps represent the number of times that a message could have been passed somewhere in the system, but not necessarily the number of times that it was. The distinction is that agents holding a message will carry it with them, possibly passing it on in later steps. This provides a means of comparison to hop-based routing statistics. It should be kept in mind, however, that some of these "hops" are used not to pass the message in the conventional sense, but to position agents so that the message can later be passed.

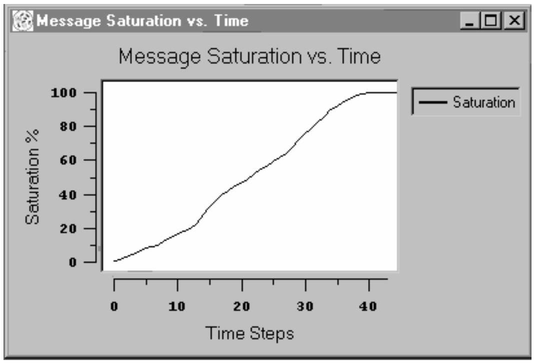

Figure 8. Typical run-time graph at 20% density

For Case 2 simulation runs the existence of ad hoc networks in the system was exploited. In these runs the message was passed to all agents that were connected to any agent having the message during a single time step.

Comparing Tables 1 and 2 shows a reduction in time steps from 109 to 88 to reach all agents in a ten percent dense grid when the ad hoc networks are used. This is a reduction of 19%. At twenty percent density, the improvement is 36%. As the density increases, the improvement seen from exploiting the networks increases as well. At fifty percent density, it reaches 70%. This is to be expected as it reflects the increasing likelihood that an agent will be within range of other agents as the grid becomes more densely populated.

Case 3 simulations examined the effects of restricting the movements of some number of agents to a fixed direction. The data in Table 3 show that this did not have much effect on message dispersal. This is because the fixed motion is still a subset of the potential random motions and that the difference is not large enough to greatly affect the outcome.

Case 4 simulations considered agents that purged the message after some number of steps then refused to accept it again. As seen in Table 4, at twenty percent density all agents would receive the message before it was lost in the system even when the message was retained for only ten time steps. With a density as low as fifteen percent, 96% of the agents could receive the message before it was lost.

Next, Case 2 was revisited and simulations were run for regions having different dimensions. All areas were rectangular and varied from a long, narrow region to a square. In each of the other sets of tests reported above the region studied was a 50 x 50 grid. These dimensions were used for the square region examined here. The other regions studied were defined in such a way that the area remained as close as possible to 2500. In each run the message originated near the center of the region.

As can be seen from the graph in Figure 6, the message in the long, narrow region (width to length ratio of 0.04) took a large number of steps to saturate the region relative to those required for a square region (W/L = 1.00). This was true regardless of the agent density. Also, regardless of the density, as the width/length ratio increased, the number of steps required for saturation dropped quickly and at around 0.25 became almost flat. This suggests that messages can be more efficiently disseminated throughout regions for which the length is at most four times the width. Since the passing of messages in the systems being studied relies on the movement of the agents, this observation can be explained by the fact that the more narrow a region is the more restricted the movement of agents in the region.

Finally, Table 6 shows an inverse relationship between the density and the number of time steps required to reach saturation. If a message is retained for a longer period, saturation can still be reached in a system with a lower density. This is not surprising but the data also show that at all density/retention combinations studied a threshold is clearly present above which full—or nearly full—saturation can be expected and below which the message will be lost from the system before reaching all agents. Specifically, on average, the message reached 20 percent of the agents at settings below the threshold while, just above it, 80 percent or more received it.

CONCLUSIONS

This paper presents a method for simulating message passing in highly mobile ad hoc networks and uses it to show the potential for the efficient broadcast of messages in these networks by treating the nodes in the network as autonomous agents. It is recognized that because the agents move on their own without any centralized control or knowledge of the system state it is not possible to guarantee that all agents will receive a message but it also appropriate to consider reaching nearly all agents to be acceptable.

It has been shown here that even at relatively low densities a message will be delivered to all (or nearly all) agents. Further, if the ad hoc networks are exploited, a significant improvement in the dispersal speed can be realized. It was also shown that if the agents purge messages after some period of time and refuse to reacquire them, the message is still efficiently dispersed even when the retention time is short.

An important consideration in any mobile message-passing scheme is keeping the redundant transmission of messages to a minimum. Ideally, of course, agents would only transmit (or relay) a given message until all agents in the system have received it. However, with no globally aware controlling entity, it is not possible to know when that total saturation point has been reached. Agents can instead set a limit on the number of time steps for which a message will be passed on. The Case 4 analysis shows that this number can be relatively low and still provide full, or nearly full, saturation.

Finally, it has been shown that the shape of the region can affect the efficiency of message saturation. This awareness can possibly be used to advantage when designing systems relying on this method of message delivery.

FUTURE WORK

This work focused on broadcast messages. A natural extension is to refine these techniques to provide point-to-point or multicast message routing. For this, the ability to return an acknowledgment would be helpful.

In the simulations reported here the agents were all of the same type. Consideration could be given to systems containing more than type of agent, such as agents with different communication ranges or speeds.

The effects of shape can be further studied. One question to consider is how the results reported here scale. That is, whether or not all square regions, say, behave similarly or whether the relative size of the area is a factor. Also, regions with non-rectangular shapes could be studied.

Finally, the scalable broadcast algorithm presented by Peng and Lu [19] could be applied to delay the start of an agent's passing of a message and the results compared to those presented here.

REFERENCES

[1] Jim Binkley and William Trost. 2001. "Authenticated Ad Hoc Routing at the Link Layer for Mobile Systems." *Wireless Networks 7*. Kluwer Academic Publishers, pp. 139–145.

[2] Jon Inouye; Jim Binkley; Jonathan Walpole. 1997. "Dynamic Network Reconfiguration Support for Mobile Computers." *Proceedings of the Third Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'97)*, pp. 13–22.

[3] Neal Singer. 2000. "Avalanche Victims Found Four Times Faster With New Sandia-Developed 'Swarm' Algorithm Technique." *Sandia Lab News*, Vol. 52, No. 2 (January 28, 2000).

[4] Mitchel Resnick. 1994. *Turtles, Termites, and Traffic Jams—Explorations in Massively Parallel Microworlds*. The MIT Press, pp. 3, 34–35.

[5] Eric Bonabeau; Marco Dorigo; Guy Theraulaz. 1999. *Swarm Intelligence, From Natural to Artificial Systems*. Oxford University Press, chapter 2.

[6] Micah Adler and Christian Scheideler. 1998. "Efficient Communication Strategies for Ad-Hoc Wireless Networks." *Proceedings of the Tenth Annual ACM Symposium on Parallel Algorithms and Architectures*, pp. 259–268.

[7] Yu-Liang Chang and Ching-Chi Hsu. 2000. "Routing in Wireless/Mobile Ad-hoc Networks via Dynamic Group Construction." *Mobile Networks and Applications 5*, pp. 27–37.

[8] Charles Perkins and Pravin Bhagwat. 1994. "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers." SIGCOMM 94, pp.234–244.

[9] Per Johansson; Tony Larsson; Nicklas Hedman; Bartosz Mielczarek; Mikael Degermark. 1999. "Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks." Mobicom '99, pp. 195–206.

[10] K. Mani Chandy and Leslie Lamport. 1985. "Distributed Snapshots: Determining Global States if Distributed Systems." *ACM Transactions on Computer Systems, Vol. 3, No. 1,* (February 1985), pp. 63–75.

[11] Amy Murphy; Gruia-Catalin Roman; George Varghese. 1997. "An Algorithm for Message Delivery to Mobile Units." PODC 97, p. 292.

[12] Charles Perkins. 1998. “Mobile Networking in the Internet.” *Mobile Networks and Applications 3,* pp.319–334.

[13] Hannes Hartenstein; Bernd Bochow; Andre Ebner; Matthias Lott; Markus Radimirsch; Dieter Vollmer. 2001. “Position-Aware Ad Hoc Wireless Networks for Inter-Vehicle Communications: the Fleetnet Project.” MobiHoc 2001, pp. 259–262.

[14] Zong Chen; H T Kung; Dario Vlah. 2001. “Ad Hoc Relay Wireless Networks over Moving Vehicles on Highways,” MobiHoc 2001, pp. 247–250.

[15] Linda Briesemeister and Gunter Hommel. 2000. “Role Based Multicast in Highly Mobile but Sparsely Connected Ad Hoc Networks.” MobiHoc 2000, pp. 45–50.

[16] Subir Kumar Das; B. S. Manoj; C. Siva Ram Murthy. 2002. “A Dynamic Core Based Multicast Routing Protocol for Ad Hoc Wireless Networks.” MobiHoc 2002, pp. 24–34.

[17] Yuanzhu Peter Chen and Arthur L. Liestman. 2002. “Approximating Minimum Size Weakly-Connected Dominating Sets for Clustering Mobile Ad Hoc Networks,” MobiHoc 2002, pp. 165–172.

[18] Brad Williams and Tracy Camp. 2002. “Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks,” MobiHoc 2002, pp. 194–205.

[19] Wei Peng and Xi-Cheng Lu. 2000. “On the Reduction of Broadcast Redundancy in Mobile Ad Hoc Networks.” MobiHoc 2000, pp. 129–130.

[20] Zygmunt J. Hass; Joseph Y. Halpern; Li Li. 2002. “Gossip-Based Ad Hoc Routing.” *Proceedings of the 21st IEEE INFOCOM*, (2002), pp. 1–10.

[21] Katia Obraczka; Kumar Viswanath; Gene Tsudik. 2001. “Flooding for Reliable Multicast in Multi-Hop Ad Hoc Networks.” *Wireless Networks 7*, pp. 627–634.

[22] Christopher Ho; Katia Obraczka; Kumar Viswanath; Gene Tsudik. 1999. “Flooding for Reliable Multicast in Multi-Hop Ad Hoc Networks.” *Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pp. 64–71.

[23] Swarm Development Group, http://www.swarm.org

[24] N. Minar; R. Burkhart; C. Langton; M. Askenzi. 1996. *The Swarm Simulation System: A Toolkit for Building Multi-Agent Simulations*, pp. 1–11.

[25] Fabrice Chantemargue; Thierry Dagaeff; Beat Hirsbrunner. 1997. "Emergence-based Cooperation in a Multi-Agent System." *Proceedings of the Second European Conference Cognitive Science (ECCS'97),* Manchester U.K., April 9-11, pp. 91–96.

[26] Sam R. Thangiah; Olena Shmygelska; William Mennell. 2001. “An Agent Architecture for Vehicle Routing Problems.” ACM Symposium on Applied Computing 2001, pp. 517–521.

[27] StarLogo, http://education.mit.edu/starlogo