

Improving Network Connectivity in Emergency Ad Hoc Wireless Networks

Abdullah Konak

Penn State Berks
aak3@psu.edu

ABSTRACT

Wireless Ad Hoc Networks (MANETs) can provide first responders and disaster management agencies with a reliable communication network in the event of a large-scale natural disaster that devastates majority of the existing communication infrastructure. Without requiring a fixed infrastructure, MANETs can be quickly deployed after a large-scale natural disaster or a terrorist attack. On the other hand, MANETs have dynamic topologies which could be disconnected because of the mobility of nodes. This paper presents a decentralized approach to maintain the connectivity of a MANET using autonomous, intelligent agents. Concepts from the social network analysis along with flocking algorithms are utilized to guide the deployment decision of agents. Unlike a basic flocking algorithm where all nodes have the same importance, network metrics are used to quantify the relative importance of nodes. Computational results are presented to demonstrate the effect of various local agent behaviors on the global network connectivity.

Keywords

Ad Hoc Networks, Swarm Intelligence, Flocking

INTRODUCTION

Recent natural disasters that devastated large geographic regions, such as Typhoon Haiyan in 2013, Indian Ocean Tsunami in 2004, and Hurricane Katrina in 2005, have shown that a reliable emergency communication system is vital for the coordination of rescue and relief activities as well as rebuilding efforts in the aftermath of natural disasters. For example, one of the main reasons for the lapses in the agency coordination, logistics, search, and rescue operations in Hurricane Katrina was attributed to the total collapse of the communication infrastructure. Hurricane Katrina damaged about 50% of the radio stations, 44% of the television stations, and more than 50,000 utility poles were destroyed in Mississippi (Holguin-Veras et al. 2007). Due to the loss of power, emergency agencies did not have a reliable means of communication to coordinate their activities. Furthermore, web-based emergency decision support systems were not accessible. Although wired or wireless communication systems are designed with resilience in mind, it is very challenging to maintain the integrity of infrastructure-based communication systems after a large-scale natural disaster which devastates the infrastructures on which they depend (e.g., power, telephone, and/or communication network backbones). A rapid deployment of communication systems for the use of first responders is crucial to minimize the loss of lives in a large-scale natural disaster. Another challenge is to provide emergency workers with a means of communication in remote geographic areas where no prior communication systems existed (Manoj and Baker 2007). In the literature, Wireless Ad Hoc Networks (MANETs), which can be quickly deployed after a large-scale natural disaster or a terrorist attack, have been proposed as a solution to address these challenges (Manoj and Baker 2007).

Unlike traditional cellular wireless networks where data packets are sent from a source node to a base node (e.g., tower), then onto the receiving node, MANETs are infrastructure-less networks that do not depend on a fixed infrastructure to route data packets. MANETs have no central base stations to relay data packets or no central network management system to keep track of the location of nodes. Instead, MANET nodes are capable of routing and retransmitting data packets (Manoj and Murthy 2004). Therefore, MANETs are instantaneous, autonomous multi-hop networks that provide service to users wherever and whenever the service is needed. Development time and cost of MANETs are also relatively low when compared to infrastructure-based networks. Because of the relative ease with which MANETs can be established, the technology has found use in situations where establishing an appropriate network infrastructure would be cost or time prohibitive, such as

Proceedings of the 11th International ISCRAM Conference – University Park, Pennsylvania, USA, May 2014
S.R. Hiltz, M.S. Pfaff, L. Plotnick, and P.C. Shih, eds.

in military, emergency, or search and rescue operations.

The flexible nature of MANETs, particularly mobility of nodes, also brings a set of unique challenges that need to be addressed to realize their full potential in emergency applications. First of all, the network topology may change unexpectedly, or the network may be disconnected because of random movements of nodes. The channel capacities of wireless links are limited and depend on the distance between nodes and environmental factors. Therefore, unexpected network bottlenecks may occur as the network topology and link capacities dynamically change. In addition, certain devices may have limited power and processing capability. All these challenges may impair the performance of a MANET.

Several approaches have been proposed in the literature to address the challenges in MANETs due to unpredictable node movements. One of the major problems is the accessibility of the centralized network services used by all nodes when the network is disconnected. This problem can be addressed by replicating network services (Akyildiz et al. 2002, Perkins and Hughes 2002, Wang and Baochun 2002, Wang and Baochun 2002, Chakrabarti and Mishra 2004) or critical data (Karumanchi et al. 1999) at multiple nodes and dynamically deploying these nodes to disconnected partitions of the network. Another problem is the delivery of data packets across disconnected network partitions. In the literature, special agent nodes are proposed to address this problem (Davis et al. 2001). Special agent nodes can buffer packets until their destination nodes are reachable. When agent nodes are connected to a network partition, they deliver their payload. Alternatively, several papers (Goyal and Caffery 2002, Li and Rus 2003) propose network topology control by modifying node trajectories or power levels (Shen et al. 2005).

Augmenting network connectivity through the use of mobile agents is suggested by several authors. In this approach, agents move along with users during the operation of the network to increase connectivity among user nodes. Ou et al. (2004) propose special relay nodes that can adjust their locations to assist disconnected network partitions. Chadrashekar et al. (2004) define the problem of achieving connectivity in disconnected ground MANETs by dynamically placing unmanned air vehicles (UAVs) which function as relay nodes. Zhu et al. (2006) also propose using UAVs equipped with communication capabilities to provide services to ground-based MANETs. Hauert et al. (2008) propose the deployment of a swarm of UAVs for search and rescue missions. During a mission time, UAVs are expected to maintain direct or indirect connection to their base-station through the MANETs that they form. Konak et al. (2011) and Dengiz et al. (2011) propose a MANET management system to improve network connectivity using agent nodes managed by a centralized network management system. In this MANET management system, a particle swarm optimization algorithm is proposed to dynamically determine new locations of agents as the topology of the network changes. This approach assumes the existence of a central network management system which is aware of the global state of the network and capable of communicating with agents at all times. However, this assumption is not realistic in many real-life MANETs.

There has been increased interest in multi-agent flocking for automated control in sensor networks and UAVs (e.g., (Badonnel et al. 2005, Schiller et al. 1993, Yang Quan et al. 2005, Olfati-Saber 2007, Mingjun et al. 2008, Hung Manh and Weihua 2009, La and Weihua 2009)). The main justification for using flocking-based algorithms for automated control is their distributed information processing ability. Konak et al. (2013) propose a flocking algorithm in which autonomous agents dynamically relocate themselves without depending on a central network management system as the topology of the network changes. This paper extends the flocking algorithm of Konak et al. (2013) by considering the relative importance of individual nodes in the flocking rules. The relative importance of a node is defined as the node's contribution to the connectivity of the network. To quantify the relative importance of nodes, two measures of network centrality are used. Preliminary results from a simulation study to determine the effectiveness of network centrality measures are reported.

PROBLEM DESCRIPTION

Let $G(t) = (N(t), E(t))$ denote a MANET with node set $N(t)$ and edge set $E(t)$ at time t . Node set $N(t)$ consists of two type of nodes, user nodes ($U(t)$) and agent nodes ($A(t)$). Users are assumed to move freely, and agents are responsible for augmenting the connectivity of the network by dynamically adjusting their locations. Edge set $E(t)$ depends on the locations of users and agents, their transmission ranges, as well as other environmental factors at time t . Let vector $\mathbf{p}_i(t) = (x_i(t), y_i(t))$ represent the position of node i at time t . Given position $\mathbf{p}_i(t)$ and transmission range R_i for each node i , edge set $E(t)$ at time t is defined as follows:

$$E(t) = \{(i, j) : i, j \in N(t), i \neq j, d_{ij} \leq \min(R_i, R_j)\} \quad (1)$$

where d_{ij} is the distance between nodes i and j such that $d_{ij} = \|\mathbf{p}_i(t) - \mathbf{p}_j(t)\|$ and $\|\cdot\|$ denotes the Euclidean norm

of a vector (i.e., $\|\mathbf{p}_i(t)\| = \sqrt{(x_i(t))^2 + (y_i(t))^2}$). The network is also assumed to be undirected (i.e., $(i, j) \equiv (j, i)$).

The overall objective of the proposed system is to maximize the connectivity of users during a mission time by dynamically locating a set of agents in response to the changes in the network topology. Let $Q(G(t))$ be the percent of the connected (directly or indirectly) user node pairs at time t . If the mission time is divided into T discrete time steps, and given that agent i can travel a maximum of V_i unit distance within a time step, the overall problem can be formulated as follows:

Problem Agent:

$$\text{Max } z = \frac{1}{T} \sum_{t=1}^T Q(G(t))$$

subject to:

$$\|\mathbf{p}_i(t) - \mathbf{p}_i(t-1)\| \leq V_i \quad \forall i \in A(t), t = 1, \dots, T$$

Problem Agent is a non-linear mixed-integer programming problem, which is very difficult to optimally solve with limited CPU resources of MANET nodes (Dengiz et al. 2011). Furthermore, the future locations of users are not known when the deployment decision is made at any time t . In the literature, the future locations of users are predicted for a planning horizon which includes several future time periods, and Problem Agent is attempted to be optimally solved within the planning horizon (Dengiz et al. 2011, Konak et al. 2011). Even with such an approximation approach, significant computing resources are required to solve Problem Agent.

The agent deployment approach introduced in this paper is a decentralized algorithm based on flocking agents (Reynolds 1987). In the proposed agent deployment approach, agents use simple rules and local information to determine their new locations as the topology of the network changes randomly. Konak et al. (2013) define a set of flocking rules to solve Problem Agent in a dynamic manner. In this paper, information about the relative importance of nodes in a network is integrated with these flocking rules to improve the network connectivity.

AGENT DEPLOYMENT ALGORITHM

Flocking is a collective behavior of independent but interacting agents. Early work on the flocking and swarm theory focused on simulating realistic movements of flocking animals. Reynolds (1987) introduced three basic rules, cohesion, separation, and alignment, to achieve the first simulated flocking behavior in computer animations. In Reynolds' model, each agent can make its independent decisions to maintain the flock. In addition, it is assumed that individual agents have no knowledge of the wider arrangement of other agents, and instead they are only aware of those agents in their immediate neighborhood. The cohesion rule encourages agents to stay close to its nearby flockmates. If an agent is too far away from its flockmates, it moves closer to them as a result of the cohesion rule. The separation rule is used to avoid collisions with nearby flockmates and to prevent clustering of agents in a small area. The separation rule causes agents to move away from their flockmates if they are too close to them. Through the alignment rule, agents adjust their velocity and the direction with their nearby flockmates. In Reynolds' model, each agent makes independent decisions based on these three rules.

The problem studied in this paper is quite different from simple flocking behavior. First of all, there is no control on the movement of users, who are free to move at their will. Nonetheless, the flocking rules can be used to guide the deployment decisions of agents to increase the connectivity of a MANET during its operation. Being a distributed algorithm, the flocking approach is particularly practical in cases where the global state of the MANET cannot be tracked and/or a central command center cannot communicate with agents. Therefore, the proposed flocking-based algorithm is very suitable for emergency response MANETs. Konak et al. (2013) first proposed using the cohesion and separation rules to guide the deployment decision of agents, and they showed that significant improvements in network connectivity could be achieved. In the flocking algorithm of Konak et al. (2013), network nodes are assumed to have equal importance in terms of their contributions to the connectivity of the network. In this paper, network connectivity measures are used to determine the relative importance of nodes and flocking rules are modified accordingly.

Let vector $\mathbf{v}_i(t) = (v_{x_i}(t), v_{y_i}(t))$ denote the velocity of agent i at time t . Velocity vector $\mathbf{v}_i(t)$ represents the direction and the distance that agent i intends to move between times t and $t+1$, and the new position of the agent at the beginning of time $t+1$ is given as follows:

$$\mathbf{p}_i(t+1) = \mathbf{p}_i(t) + \frac{\min(V_i, \|\mathbf{v}_i(t)\|)}{\|\mathbf{v}_i(t)\|} \mathbf{v}_i(t) \quad (2)$$

Equation (2) states that agent i is allowed to move a maximum of V_i unit distance at the direction of $\mathbf{v}_i(t)$.

Velocity vector $\mathbf{v}_i(t)$ is updated based on three rules, separation, cohesion, and exploration. The separation and cohesion rules of an agent depend on the spatial neighbors of the agent. If a node is directly connected to an agent, the node is considered as a spatial neighbor of the agent. The set of spatial neighbors of agent i ($N_i(t)$) at time t is defined as follows:

$$N_i(t) = \{j \in N(t) : d_{ij} < \min(R_i, R_j), i \neq j\}$$

Set $N_i(t)$ consists of two disjoint types of spatial neighbors, user neighbors ($NU_i(t)$) and agent neighbors ($NA_i(t)$). This distinction between user and agent neighbors is made so that agents can have different interactions with users and other agents. Each agent aims to maintain a target distance S_A with its agent neighbors and a target distance S_U with its user neighbors. Parameters S_A and S_U should be determined based on the communication range, the terrain, and the density of the network.

If the distance between an agent and one of its neighbors is larger than the target distance, then the cohesion rule is applied to the movement decision of the agent with respect to this neighbor. At time t , the velocity decision of agent i due to its cohesion interaction with its neighbor node j (i.e., $\mathbf{vc}_{ij}(t) = (vx_{ij}(t), vy_{ij}(t))$) is defined as follows:

$$\mathbf{vc}_{ij}(t) = \begin{cases} 0.5w_{jt} \max((d_{ij} - S_A), 0)(\mathbf{p}_j(t) - \mathbf{p}_i(t)) / d_{ij} & j \in NA_i(t) \\ w_{jt} \max((d_{ij} - S_U), 0)(\mathbf{p}_j(t) - \mathbf{p}_i(t)) / d_{ij} & j \in NU_i(t) \end{cases} \quad (3)$$

where w_{jt} is the relative importance of node j in the network. If distance d_{ij} in equation (3) is larger than the target distance, then agent i will move closer to node j . Note that if node j is an agent (i.e., $j \in NA_i(t)$), node j is also expected to exhibit the same movement with respect to agent i . Therefore, the velocity is multiplied by 0.5 in the first part of equation (3). Parameter w_{jt} is dynamically determined based on the topological importance of node j at time t .

If the distance between an agent and one of its neighbors is smaller than the target distance, then the separation rule is applied. The separation rule is the opposite of the cohesion rule. At time t , the velocity decision of agent i due to its separation interaction with its neighbor node j (i.e., $\mathbf{vs}_{ij}(t) = (vx_{ij}(t), vy_{ij}(t))$) is defined as follows:

$$\mathbf{vs}_{ij}(t) = \begin{cases} 0.5w_{jt} \max((S_A - d_{ij}), 0)(\mathbf{p}_i(t) - \mathbf{p}_j(t)) / d_{ij} & j \in NA_i(t) \\ w_{jt} \max((S_U - d_{ij}), 0)(\mathbf{p}_i(t) - \mathbf{p}_j(t)) / d_{ij} & j \in NU_i(t) \end{cases} \quad (4)$$

Because users move randomly, an agent or a group of agents can be disconnected from other nodes. In such cases, the separation and/or cohesion rules are not applicable, and agents move randomly with an expectation of discovering other nodes. Therefore, this rule is called the exploration rule. The velocity of agent i due to exploration rule is given as follows:

$$\mathbf{ve}_i(t) = (V_i \cos(\text{Rand}(0, 2\pi)), V_i \sin(\text{Rand}(0, 2\pi))) \quad (5)$$

where $\text{Rand}(a, b)$ is the uniform random variable between a and b . The final velocity decision of agent i at time t is given as follows:

$$\mathbf{v}_i(t) = \begin{cases} \alpha \left(w_c \sum_{j \in NA_i(t)} \mathbf{vc}_{ij}(t) + w_s \sum_{j \in NU_i(t)} \mathbf{vs}_{ij}(t) + w_e \mathbf{ve}_i(t) \right) + (1 - \alpha) \mathbf{v}_i(t-1) & N_i(t) \neq \emptyset, \\ \alpha \mathbf{ve}_i(t) + (1 - \alpha) \mathbf{v}_i(t-1) & N_i(t) = \emptyset. \end{cases} \quad (6)$$

where α is a smoothing parameter (between 0 and 1) which indicates how much previous information gained by an agent is incorporated into its new deployment decision. Note that current velocity $\mathbf{v}_i(t-1)$ of agent i may have been formed by its previous separation or cohesion interactions. Therefore, the current velocity of an agent is likely to include important information regarding its previous interactions. Smoothing parameter α can be considered as a memory parameter. A high value of α emphasizes the current interactions of the agent in the deployment decision. On the other hand, a low value of α emphasizes its previous interactions. Parameters w_s , w_c , and w_e represent the relative weights (between 0 and 1) of the separation, cohesion, and exploration rules in

the final deployment decision, respectively. It is possible to model various agent behaviors by setting these weight parameters accordingly.

Calculation of the Relative Importance of Nodes

In equations (3) and (4), parameter w_{jt} represents the relative importance of node j at time t . In the flocking-based algorithm described above, it is assumed that the global state of the network is not available to agents, and agents are allowed to interact only with its immediate neighbors. In addition, it is assumed that there is no central network management system. Therefore, it is infeasible to use a metric based on the global topology of the network to evaluate the relative importance of nodes. In this paper, two measures, degree centrality and betweenness centrality, are studied. In social network analysis, degree centrality and betweenness centrality are used to quantify the relative importance or influence of nodes in a social network.

Degree centrality is defined as the number of edges incident to a node. Based on degree centrality, the relative importance of node j is calculated as follows:

$$w_{jt} = 1 + \frac{|N_j(t)|}{\max_{k \in N_j(t)} \{|N_k(t)|\}} \quad (7)$$

where $|N_j(t)|$ represents the cardinality of set $N_j(t)$. Equation (7) requires that nodes communicate their degree centrality with their neighbors. This requirement is straightforward to achieve in real-life operations of MANETs. Therefore, w_{jt} can be efficiently calculated.

Betweenness centrality is a measure of the number of times that a node appears in the shortest paths from all nodes to all other nodes. A node that interconnects many other nodes is expected to have a high value of betweenness centrality. The relative importance of node j based on betweenness centrality is calculated as follows:

$$w_{jt} = 1 + \frac{bc_j(t)}{\max_{k \in N_j(t)} \{bc_k(t)\}} \quad (8)$$

where $bc_j(t)$ is the betweenness centrality of node j at time t . To calculate the betweenness centrality of node j , first the shortest paths between all pairs of nodes are determined and then the paths including node j are counted. In practical implementations of MANETs, $bc_j(t)$ can be estimated using the routing table of nodes.

The overall agent deployment algorithm of agent i is as follows:

```

Do Until Stop {
  Determine  $N_i(t)$ ,  $NA_i(t)$ , and  $NU_i(t)$  at its current location  $\mathbf{p}_i(t)$  for each agent  $i$ 
  Using a centrality measure to calculate  $w_{jt}$  for  $j \in N_i(t)$ 
  Calculate new velocity  $\mathbf{v}_i(t)$  using equation (6)
  Calculate the new location  $\mathbf{p}_i(t+1)$  using equation (2)
  Move agent  $i$  to new location  $\mathbf{p}_i(t+1)$ 
   $t=t+1$ 
}

```

COMPUTATIONAL EXPERIMENTS

The performance of the flocking-based algorithm and the impact of using node weights in the flocking-based algorithm on network connectivity were evaluated using simulation. Two questions were addressed in the simulation study: (i) How much improvement could be achieved by incorporating information about the relative importance of nodes into the flocking-based algorithm? and (ii) What is the performance of the flocking-based algorithm compared to other approaches? In the simulation study, users were assumed to move randomly, representing the chaotic environment during a large-scale natural disaster. In addition, random mobility models are preferred in the comparison of various MANET protocols in order to avoid any bias due to user mobility patterns.

Description of the Simulation Environment

The simulation environment was coded in Java. The flowchart of the simulation code is illustrated in Figure 1. In the simulation, first agents are deployed to their new locations (Agent Loop) based on the current state of the

network. Then, users move to their new locations which are randomly determined using a version of Random Waypoint Mobility Model (Broch et al. 1998) (User Loop). The simulation area is a circle with a radius of 300 unit distance. The initial locations and initial velocities of all nodes are randomly determined. In each simulation step, user i changes its direction randomly with a probability of ρ as follows:

$$\theta_i(t) = \begin{cases} \text{Rand}(0, 2\pi) & \text{Rand}(0, 1) \leq \rho, \\ \theta_i(t-1) & \text{otherwise.} \end{cases}$$

Then, user i travels a random distance between V_{\min} and V_{\max} in the direction of angle $\theta_i(t)$. The new position of user i is given by

$$\mathbf{p}_i(t+1) = \mathbf{p}_i(t) + \text{Rand}(V_{\min}, V_{\max})(\cos(\theta_i(t)), \sin(\theta_i(t))) \quad (9)$$

If new position $\mathbf{p}_i(t+1)$ is outside of the simulation area, $\theta_i(t)$ is randomly regenerated until it is within the simulation area. At the end of termination of the simulation, the average percent connectivity of the network is calculated as follows:

$$P = \frac{1}{T - T_1} \sum_{t=T_1}^T Q(G(t)) \quad (10)$$

where T_1 is the simulation warm-up period.

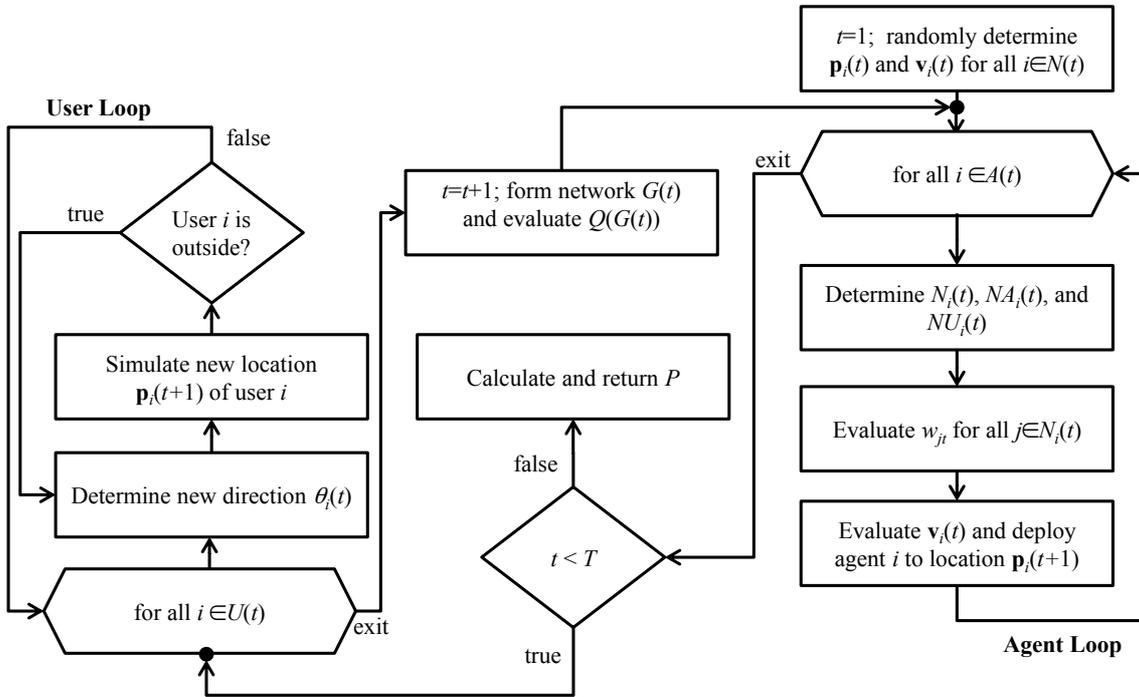


Figure 1. Flow chart of the simulation procedure

Relative importance of nodes

Thirty different network configurations, including all combinations of 10, 20, 30, 40, 50, 60 users and 3, 4, 5, 6, and 7 agents, were simulated for three scenarios depending on the method used to determine the relative importance of nodes: (i) the same weight ($w_{ji}=1$), (ii) betweenness centrality, and (iii) degree centrality. For each test network and scenario combination, 100 random simulation replications were performed with parameters: $w_s=1$, $w_c=1$, $w_e=1$, $\alpha=90$, $R_f=100$, $V_f=10$, $S_d=75$, $S_t=50$, and $\rho=0.1$. The simulation was run for $T=1000$ with warm-up period of $T_1=50$. Table 1 summarizes the averages of the percent network connectivity observed in 100 replications. The simulation results of the three scenarios were compared using a paired t -test as the same random number set was used in a random replication to simulate the scenarios (i.e., the common random number variance reduction technique). As seen in Table 1, using betweenness centrality to assign relative importance of nodes provided the highest network connectivity level in all networks. Although the differences between the scenarios with no weight and with betweenness centrality were very small (about 1%),

they were statistically significant for all networks excluding network (10, 7). As discussed in the next section, increasing network connectivity even 1% is very challenging for Problem Agent. Using node degree centrality provided the lowest network connectivity in all test cases.

The results in this study suggest that the nodes that function as a bridge between other nodes should have relatively high weight in the deployment decisions of agents. Betweenness centrality is a good importance measure of a node in terms of the node's being a bridge between other nodes. On the contrary, nodes that are already connected to other nodes with many links should have relatively low weights in the deployment decision.

Test network ($ A(t) , U(t) $)	(i) Same Node Weight	(ii) Betweenness Centrality	(iii) Node Degree Centrality	Test network ($ A(t) , U(t) $)	(i) Same Node Weight	(ii) Betweenness Centrality	(iii) Node Degree Centrality
(10, 3)	19.3	19.6	18.6	(40, 3)	45.9	46.6	45.5
(10, 4)	21.1	21.4	19.9	(40, 4)	47.4	48.2	46.7
(10, 5)	22.8	23.1	21.4	(40, 5)	48.8	49.8	48.0
(10, 6)	24.4	24.5	22.5	(40, 6)	50.2	51.3	49.0
(10, 7)	25.9	26.0 ^(a)	23.7	(40, 7)	51.6	52.7	50.2
(20, 3)	24.0	24.7	23.5	(50, 3)	60.9	61.3	60.6
(20, 4)	25.8	26.4	24.9	(50, 4)	62.4	62.7	61.7
(20, 5)	27.2	28.1	26.1	(50, 5)	63.5	64.1	62.7
(20, 6)	28.8	29.5	27.2	(50, 6)	64.7	65.3	63.7
(20, 7)	30.1	31.1	28.2	(50, 7)	65.9	66.6	64.6
(30, 3)	33.4	34.1	32.9	(60, 3)	74.2	74.4	73.9
(30, 4)	34.9	35.7	34.3	(60, 4)	75.2	75.4	74.8
(30, 5)	36.5	37.5	35.5	(60, 5)	76.2	76.4	75.6
(30, 6)	38.0	39.0	36.7	(60, 6)	77.1	77.4	76.2
(30, 7)	39.4	40.6	37.7	(60, 7)	77.9	78.2	77.0

Table 1. Effectiveness of the alternatives ways to assign relative importance of node weights ^(a) the difference between (i) and (ii) is not statistically significant with $p < 0.05$)

Comparative performance of the flocking algorithm

In this section, the flocking-based algorithm is compared with an exact approach to solve Problem Agent for a single time period. Konak et al (2013) describes a mathematical model to determine the optimal locations of agents based on the global state of the network at time t . In Figure 2, the performance of the flocking-based algorithm is compared with a mathematical programming approach in which Problem Agent was solved to optimality for each time period. In the simulation, the Gurobi Mixed-Integer Quadratically Constrained Programming solver was used to solve Problem Agent optimally to determine the new locations of agents at each time period. It should be noted that agent deployments determined by Problem Agent were not optimal for the entire mission time because the optimization in each time period was considered independently for the tractability of the problem. In addition to the mathematical programming, the flocking-based algorithm was compared with the total random behavior of the agents (i.e., $w_s=0$, $w_c=0$, and $w_e=1$). The random behaviors of agents provide a basis to gauge the performance of the flocking-based algorithm. The same simulation parameters were used with the previous study, but the parameter of the flocking heuristic was set as follows: $w_s=0.5$, $w_c=0.75$, and $w_e=0$ for networks with 10 users; $w_s=0.75$, $w_c=0.5$, and $w_e=0$ for networks with 20 users; and $\alpha=0.90$, $S_A=75$, $S_U=50$ for all networks. Because of the extensive simulation time required when mathematical programming was used, larger size networks could not be simulated, and only thirty random simulation replications were performed for each test network.

As seen in Figure 2, the performances of the flocking-based algorithm and the mathematical programming were very close. When the results of the flocking-based algorithm and mathematical programming approach were compared using a t -test, the performances of these two approaches were not statistically different for all test networks, excluding test network (20, 2). In several test networks, the flocking-based algorithm provided slightly higher levels of connectivity than the mathematical programming approach. As stated earlier, an agent deployment policy determined by the mathematical programming approach was not optimal for the entire simulation time because time periods were considered independently. The best deployment decision at a time

period may have had adverse effects for future periods. Comparing to the random behavior of agents, the flocking-based algorithm and mathematical programming provided about 3% to 5% higher network connectivity. Although such a level of improvement seems low, it should be noted that Problem Agent is a very challenging problem to solve, particularly in a dynamic manner within a short time period. The flocking-based algorithm is a very straightforward and promising approach to solve this difficult dynamic optimization problem in a distributed manner with limited computing resources. The results in this section also show that 1% improvement achieved by considering node weights based on betweenness centrality is difficult to achieve.

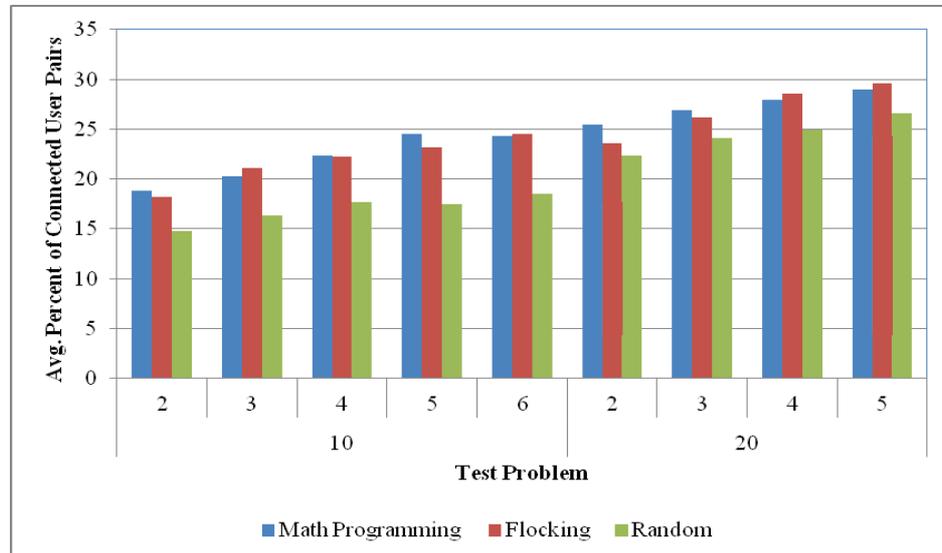


Figure 2. Comparative performance of the flocking-based algorithm

CONCLUSIONS

This paper demonstrated how relative importance of nodes could be incorporated into a flocking-based algorithm to improve connectivity in MANETs. Due to the dynamic nature of MANETs, agent deployment decisions must be made dynamically, in a short time, and based on incomplete information about the global state of the network. Current approaches to the problem are limited by their dependency on a centralized network management system. The proposed flocking-based algorithm has a minimum overhead and can be implemented in a decentralized and asynchronous manner. Therefore, the proposed approach can be used to maintain connectivity in emergency MANETs where a central network management system does not exist. The simulation experiments have shown that the proposed flocking-based algorithm is promising. Particularly, using node weights in the proposed flocking-based algorithm is effective for improving network connectivity. Further research is required to develop an adaptive strategy to determine node weights and the parameters of the flocking-based algorithm.

REFERENCES

1. Akyildiz, I., W. Su, Y. Sankarasubramaniam and E. Cayirci (2002). "Wireless sensor networks: a survey." *Computer Networks*, **38**(4), 393-422.
2. Badonnel, R., R. State and O. Festor (2005). "Self-organized monitoring in ad-hoc networks." *Telecommunication Systems - Modeling, Analysis, Design and Management*, **30**(1-3), 143-160.
3. Broch, J., D. A. Maltz, D. B. Johnson, Y. C. Hu and J. Jetcheva (1998). *A performance comparison of multi-hop wireless ad hoc network routing protocols*, New York, NY, USA, ACM.
4. Chadrashekar, K., M. R. Dekhordi and J. S. Baras (2004). Providing Full Connectivity in Large Ad-Hoc Networks by Dynamic Placement of Aerial Platforms. *The Center for Satellite and Hybrid Communication Networks, University of Maryland*, University of Maryland.
5. Chakrabarti, S. and A. Mishra (2004). "Quality of service challenges for wireless mobile ad hoc networks." *Wireless Communications and Mobile Computing*, **4**(2), 129-153.
6. Davis, J. A., A. H. Fagg and B. N. Levine (2001). *Wearable computers as packet transport mechanisms in highly-partitioned ad-hoc networks*. Proceedings of Fifth International Symposium on Wearable Computers, Los Alamitos, CA, USA, IEEE Comput. Soc.

Proceedings of the 11th International ISCRAM Conference – University Park, Pennsylvania, USA, May 2014
S.R. Hiltz, M.S. Pfaff, L. Plotnick, and P.C. Shih, eds.

7. Dengiz, O., A. Konak and A. E. Smith (2011). "Connectivity management in mobile ad hoc networks using particle swarm optimization." *Ad Hoc Networks*, **9**(7), 1312-1326.
8. Goyal, D. and J. Caffery Jr (2002). *Partitioning avoidance in mobile ad hoc networks using network survivability concepts*. Proceedings of Seventh IEEE Symposium on Computers and Communications (ISCC'02), Taormina-Giardini Naxos, Italy
9. Hauert, S., L. Winkler, J.-C. Zufferey and D. Floreano (2008). "Ant-based swarming with positionless micro air vehicles for communication relay." *Swarm Intelligence*, **2**(2-4), 167-188.
10. Holguín-Veras, J., N. Pérez, S. Ukkusuri, T. Wachtendorf and B. Brown (2007). "Emergency logistics issues affecting the response to Katrina: a synthesis and preliminary suggestions for improvement." *Transportation Research Record: Journal of the Transportation Research Board*, **2022**(1), 76-82.
11. Hung Manh, L. and S. Weihua (2009). *Adaptive flocking control for dynamic target tracking in mobile sensor networks*. 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009), Piscataway, NJ, USA, IEEE.
12. Karumanchi, G., S. Muralidharan and R. Prakash (1999). *Information dissemination in partitionable mobile ad hoc networks*. Proceedings of the 18th IEEE Symposium on Reliable Distributed Systems, Los Alamitos, CA, USA, IEEE Comput. Soc.
13. Konak, A., G. Buchert and J. Juro (2013). "A flocking-based approach to maintain connectivity in mobile wireless ad hoc networks." *Applied Soft Computing*, **13**(2), 1284-1291.
14. Konak, A., O. Dengiz and A. E. Smith (2011). "Improving Network Connectivity in Ad Hoc Network Using Particle Swarm Optimization and Agents", *Wireless Network Design: Optimization Models and Solution Procedures (International Series in Operations Research and Management Sciences)*. J. Kennington, E. Olinick and D. Rajan, Springer **158**: 247-267.
15. La, H. M. and S. Weihua (2009). *Flocking control of a mobile sensor network to track and observe a moving target*. 2009 IEEE International Conference on Robotics and Automation (ICRA), 12-17 May 2009, Piscataway, NJ, USA, IEEE.
16. Li, Q. and D. Rus (2003). "Communication in disconnected ad hoc networks using message relay." *Journal of Parallel and Distributed Computing*, **63**(1), 75-86.
17. Manoj, B. and C. Murthy (2004). *Ad Hoc Wireless Networks: Architectures and Protocols*, Prentice Hall.
18. Manoj, B. S. and A. H. Baker (2007). "Communication challenges in emergency response." *Communications of the ACM*, **50**(3), 51-53.
19. Mingjun, X., H. Liusheng, H. He and W. Pei (2008). *Swarm intelligence: coverage and connectivity control for mobile sensors*. 2008 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Piscataway, NJ, USA, IEEE.
20. Olfati-Saber, R. (2007). *Distributed tracking for mobile sensor networks with information-driven mobility*. 2007 American Control Conference, Piscataway, NJ, USA, IEEE.
21. Ou, C. H., K. F. Ssu and H. C. Jiau (2004). *Connecting network partitions with location-assisted forwarding nodes in mobile ad hoc environments*. 10th Pacific Rim International Symposium on Dependable Computing (PRDC'04), Papeete, Tahiti, French Polynesia.
22. Perkins, D. D. and H. D. Hughes (2002). "A survey on quality-of-service support for mobile ad hoc networks." *Wireless Communications and Mobile Computing*, **2**(5), 503-513.
23. Reynolds, C. W. (1987). *Flocks, herds and schools: a distributed behavioural model*. Proceedings of the Conference held at Computer Graphics 87, Pinner, UK, Online Publications.
24. Schiller, I., J. S. Luciano and J. Stark Draper (1993). *Flock autonomy for unmanned vehicles*. SPIE - Int. Soc. Opt. Eng. (USA), USA.
25. Shen, C. C., Z. Huang and C. Jaikao (2005). "Ant-based distributed topology control algorithms for mobile ad hoc networks." *Wireless Networks*, **11**(3), 299-317.
26. Wang, K. H. and L. Baochun (2002). *Efficient and guaranteed service coverage in partitionable mobile ad-hoc networks*. Proceedings of IEEE INFOCOM 2002 Conference on Computer Communications. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.37364), Piscataway, NJ, USA, IEEE.
27. Wang, K. H. and L. Baochun (2002). *Group mobility and partition prediction in wireless ad-hoc networks*. Proceedings of 2002 IEEE International Conference on Communications, Piscataway, NJ, USA, IEEE.
28. Yang Quan, C., W. Zhongmin and L. Jinsong (2005). *Automatic dynamic flocking in mobile actuator sensor networks by central Voronoi tessellations*. 2005 IEEE International Conference on Mechatronics and Automation (IEEE Cat. No. 05EX1044), Piscataway, NJ, USA, IEEE.
29. Zhu, H., A. L. Swindlehurst and K. J. R. Liu (2006). *Smart deployment/movement of unmanned air vehicle to improve connectivity in MANET*. 2006 IEEE Wireless Communications and Networking Conference, Vegas, NV, USA.