A Stochastic Approach for Topology Management of Mobile Ad-hoc Networks

Sourav Sen Gupta, Surjya S. Ray, Somnath Samanta, Oly Mistry and Mrinal K. Naskar
Electronics and Tele-Communication Engineering Department
Jadavpur University, Kolkata 700032
E-mail: *damal2003@yahoo.com*

Abstract

In this paper, we propose a probabilistic topology management algorithm for MANET. The basic philosophy of the algorithm is to elect a group leader from the nodes and endow it with the responsibility of managing the network neighborhood topology, thereby allowing the nodes to communicate with each other without any requirement of routers. The group leader uses the stochastic properties of the nodes' random waypoint mobility model to calculate the probabilities of divergence of the nodes from a predefined zone of stability. The zone of stability is defined in such a fashion that any two nodes within that zone can communicate directly with each other. Next, based on some predefined threshold probability, it controls the motion of the nodes to manage the neighborhood topology, thereby ensuring that the connectivity of one node with other will not change during the movement of the network. Hence, the routing overhead may completely be eliminated. We have simulated our algorithm through a number of synthetically designed situations using various threshold probabilities and have obtained encouraging results.

1. Introduction

A Mobile Ad-Hoc Network (MANET) is a decentralized network of autonomous mobile nodes able to communicate over wireless links without the help of any fixed infrastructure [1,2]. In such network mobile nodes connect dynamically in an arbitrary fashion. These nodes behave as usual trans-receivers and also as routers, taking part in discovery and maintenance of routes to other nodes in the network. MANET is projected to play a vital role in applications like search-and-rescue operations, meetings and conventions, mobile sensors or satellite networks where quick sharing of information or data acquisition is required in inhospitable terrain.

Mobile nodes may move frequently and it may so happen that when a node wants to transmit data to another node, the intended receiver may fall outside the range of communication of the transmitting node. This makes routing an essential requirement in MANET. There are many existing routing schemes for MANET that can be divided into four types namely flooding, proactive routing, reactive routing and dynamic cluster based routing [3].

Flooding based routing requires no knowledge of the network topology. Although such protocols are effective under light load conditions, they generate excessive amount of traffic for large networks. This makes it difficult to achieve flooding reliably [3]. Proactive routing protocol is basically a table driven routing protocol where each node pre-computes the route to every possible destination as well as the path to be followed to minimize the cost. The protocol also periodically broadcasts routing information throughout the network. This approach however increases network traffic in highly dynamic networks. Several modified proactive routing protocol have been suggested [3] to minimize the traffic. M.Joa-Ng and I.T.Lu proposed a zone based routing protocol [4] where the network is divided into several non-overlapping zones. Reactive routing is a very lazy on-demand approach in the sense that it takes a long time to find a route from source to destination and uses query-response mechanism to find the route. Ad-hoc on-demand distance vector (AODV) routing is a good example of reactive routing. In this approach overhead may increase significantly due to frequent route finding for highly dynamic networks. Several reactive routing protocols have been proposed so far. C.K.Toh proposed association based reactive routing [5] to find a stable route. The temporary ordered routing algorithm (TORA) is another reactive routing scheme where multiple nodes from source to destinations are calculated by localizing the control messages to a very small set of nodes. The main drawback of reactive approach compared to proactive routing is the significant delay of route setup time and also the large volume of control traffic, which is required to support route query mechanism. In dynamic cluster based routing protocol [6] the network is dynamically organized into partitions known as clusters to maintain a stable effective topology. Several clustering algorithms are also proposed, which differ from one another in the criteria used to organize the clusters, such as prediction of node mobility etc. However, none of the proposed schemes guarantees constant network connectivity during the movement and each of these schemes have constant route maintenance overhead. A particular node may even be disconnected in the worst case.

To avoid this disconnection problem, we had proposed a few centralized topology management algorithms in our prior works [7,8]. In the present case, however, we are utilizing the Global Positioning System facility for better topology retention. In urban setting, this scheme can easily be modified using Bluetooth or other positioning utilities.

In this paper, we suggest a self-adaptive movement control algorithm of mobile nodes to ensure the retention of network connectivity even during the positional variation of the nodes. The key concept is to elect a movement coordinator in the network to direct the movement of the other nodes based on the position and velocity information of the nodes obtained using the GPS. The Coordinator directs the nodes to move in such a fashion that the distance from any node to the coordinator does not exceed a predefined maximum value. This maximum range of movement will ensure that, two particular nodes which were neighboring nodes at the beginning will remain so during the movement too. Eventually, the path between any two nodes will not change throughout the entire movement and hence the routing overhead can be eliminated.

The paper is organized as follows. In the next section, we formally define the topology management problem. In the third section, we present the proposed algorithm for maintaining the topology. The fourth section contains the necessary mathematical framework for the stochastic approach. The lemmas along with their proofs have been included in the fifth section. The sixth section presents the simulation results obtained for a number of synthetically designed scenarios and for different values of the threshold probability. This section also presents a comparative study between the algorithm proposed in our previous work [8] and our new algorithm. Finally, section 7 concludes the paper.

2. Topology Management Problem

Given a physical topology of a mobile ad-hoc network and the directions of movement of the nodes which are assumed to be unidirectional, the problem is to control the movements of the individual nodes so as to maintain a stable neighborhood topology. The objective is to allow the nodes to communicate amongst themselves without the need of any routing.

Let us consider a MANET consisting of N number of mobile nodes $n_0, n_1, n_2 \ldots n_{N-1}$. Let us also assume that each node of the network has a maximum transmission range of R_{max}. Now, any two nodes n_i and n_j are called neighboring nodes if they can communicate amongst themselves without the help of any routing. So, the two nodes will be neighbors if and only if $D(i, j) \leq R_{max}$ where $D(i, j)$ is defined as the relative distance between the nodes n_i and n_j. The network neighborhood topology will be maintained if and only if:

$$D(i, j) \leq R_{\max} \quad \forall \; i, j = 0, 1, 2 \ldots N-1$$

3. The Stochastic Algorithm

The proposed algorithm maintains neighborhood topology in a mobile ad-hoc network through restricted movements of all the mobile nodes using message communication. The basic philosophy of this algorithm is to elect a movement coordinator, which controls the movements of the other nodes, based on the probabilistic aspects of their random waypoint mobility model, to maintain the network neighborhood topology. The algorithm is divided into two parts, namely Coordinator Election Algorithm and Movement Algorithm.

Each individual node in this MANET is a mobile trans-receiver provided with a GPS receiver. In this scheme, a node can also vary the transmission range stepwise whenever required. The algorithm prescribes two transmission ranges, namely Short Range (R_{min}) and Long Range (R_{max}).

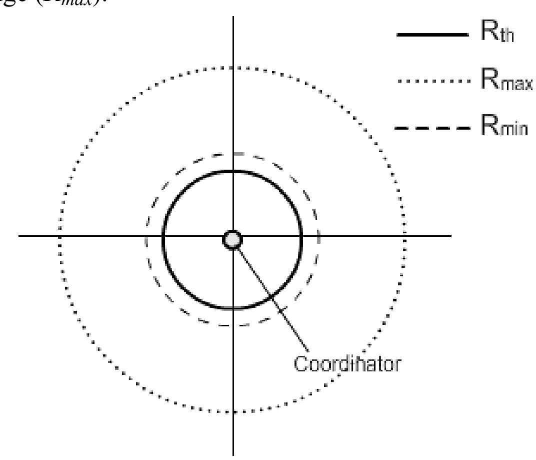


Figure 1: Communication ranges and the threshold range

The algorithm defines four types of control messages, namely POSITION-message, START-message, STOP-message and RUSH-message. The POSITION message is sent to the Coordinator by all the nodes at regular interval of T/2, where the time interval T is the time to cross the distance between a predefined threshold range R_{th} and R_{min} with the predefined maximum velocity V_{max}. Mathematically: $T = \left(R_{\min} - R_{th}\right) / V_{\max}$

The POSITION message contains the positional coordinates of the node, obtained from the GPS. The START message is used to instruct the nodes to move with their preferred velocities, while the STOP message instructs the nodes to halt until the next instruction. The RUSH message instructs the nodes to rush for an interval of T/2.

The short range of communication, R_{min}, is used by the nodes to send the POSITION message to the coordinator and it is used by the coordinator to transmit the control instructions like START, STOP or RUSH for controlling the movement of the nodes. The long range of communication, R_{max}, is the one used for transmitting actual data packets.

3.1. Zone of Stability

We call a node to be stable if it can pursue its normal movement with its own preferred velocity, and can transmit data packets directly to any other node within the network. We define the stability of the whole network with respect to the position of the coordinator. If all the nodes of the network are stable with respect to the coordinator, we call the whole network to be stable.

The network is said to be unstable if any one of the nodes becomes unstable. As per the proposed algorithm, the coordinator tries to stabilize that node by controlling the movement of the whole network. If all the nodes of the network reside within a particular region, defined as the Zone of Stability, then the network can be stabilized within a finite amount of time by movement control. The Zone of stability is illustrated in the figure 2 below.

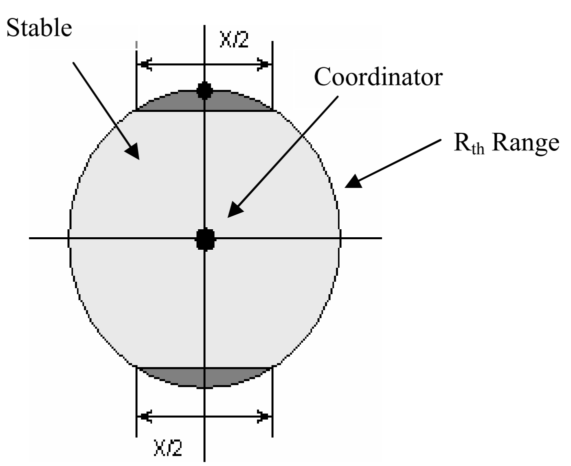


Figure 2: The zone of stability

Here, the circle in the figure depicts the Threshold Range R_{th} and X is the radial distance between Short Range and Threshold Range. Therefore, $X = R_{\min} - R_{th}$. The Zone of stability selected by virtue of Lemma 3.

3.2. Assumptions

The algorithm is based on some assumptions. They are:

1. All the nodes of the network must move in the same direction.
2. All nodes should have a predefined maximum velocity $V_{max.}$
3. Acceleration and deceleration of the nodes are taken to be instantaneous.
4. It is assumed that if the nodes are within appropriate range, the instruction messages will never be lost in transit.
5. At the beginning, there must be at least one node whose distance from all the other nodes is within the zone of stability.
6. Each node is provided with a unique identification number.
7. All nodes have a GPS unit that gives approximate 3-dimensional position, velocity and accurate time in Universal Time Coordinate (UTC) format.
8. The information as received from GPS is assumed to be correct.

3.3. Coordinator Election Algorithm

Any node, whose distance from all other nodes in the network is within the stable zone as defined above, is eligible for being the coordinator. The eligible node having the least identification number is chosen as the coordinator in case of any conflict. An extra message namely CONFLICT is defined to elect the coordinator.

At the beginning, all nodes broadcast a POSITION message to all other nodes using R_{min}. Each node calculates the distance of all the other nodes from itself. If all the other nodes are within the Zone of Stability with respect to a node, it becomes eligible for being the coordinator.

If multiple nodes are eligible for being the coordinator, the conflict situation arises. In this case, all the nodes, which are eligible to be a coordinator, send a CONFLICT message to all other nodes along with the identification number, and thus all the nodes come to know about the eligibility of the other nodes. The eligible node that finds that all the other eligible nodes have numbers greater than it declares itself as the coordinator. Then it initiates the movement of the network.

3.4. Movement Algorithm

Once the coordinator is elected, it sends a START message to all other nodes in the network. After each node receives the START message the network starts moving. The movement algorithm is as follows:

1. The Coordinator receives the POSITION message from each node.
2. The Coordinator calculates the distance D_i of each node n_i from itself.
3. It checks if any node has diverged out of the Zone of Stability.
4. If any node has diverged AHEAD of it, the coordinator instructs that node to STOP and the coordinator along with all the well-connected nodes RUSH for the subsequent T/2 interval.
5. If on the other hand, any node has diverged BEHIND it, the coordinator instructs that node to RUSH and the coordinator along with all the well-connected nodes STOP for the subsequent T/2 interval.
6. If some nodes have diverged AHEAD as well as some have diverged BEHIND, the coordinator first instructs those which are behind to RUSH and all the other nodes (including those AHEAD) to STOP. After the next T/2 interval, the nodes which were BEHIND will surely converge (Lemma 2). Now, the coordinator will instruct the well-connected nodes to RUSH with itself for the next time interval T/2 while those AHEAD are instructed to stop. This will converge the nodes which were AHEAD (Lemma 2).
7. If all the nodes are within the Zone of Stability, the coordinator calculates the Probability of Divergence (P_{div}) of each node from the Zone of Stability (the mathematical derivation of the probability is explained in section 4).
8. It compares this Divergence Probability (P_{div}) with a predefined Threshold Probability (P_{th}).
9. If $P_{div} \leq P_{th}$, then the Coordinator does not take any action for that node. Else, if the node is ahead of the coordinator, it is instructed to STOP and if the node is behind the coordinator, it is instructed to RUSH. All the other nodes continue their normal movement with their preferred velocities.

```
Common Data Structure: Var STATUS: array [0  N-1] of logical
                       Var NODES_AHEAD, NODES_BEHIND: integer

Initialization: STATUS[i] ←  WITHIN;
                NODES_AHEAD ← 0;
                NODES_BEHIND ← 0;

Coordinator receives the POSITION messages from all the other nodes;

For i:=0 to N-1 do
  Calculate distance from coordinator D_i = sqrt((x_i - x_o)^2 + (y_i - y_o)^2);
  If D_i ≤ R_th
    Continue;
  Else
    If the node is ahead of the coordinator
      STATUS[i] = AHEAD;
      NODES_AHEAD = NODES_AHEAD + 1;
    Else If the node is behind the coordinator
      STATUS[i] = BEHIND;
      NODES_BEHIND = NODES_BEHIND + 1;
    End If;
  End If;
End For;

If NODES_AHEAD ≠ 0  &&  NODES_BEHIND = = 0
  Send STOP message to all nodes n_i with STATUS[i] = AHEAD;
  Send RUSH message to all nodes n_i with STATUS[i] = WITHIN;

Else If NODES_AHEAD = = 0  &&  NODES_BEHIND ≠ 0
  Send STOP message to all nodes n_i with STATUS[i] = WITHIN;
  Send RUSH message to all nodes n_i with STATUS[i] = BEHIND;
```

```
Else If NODES_AHEAD ≠ 0  &&  NODES_BEHIND ≠ 0
   Send STOP message to all nodes n_i with STATUS[i] = AHEAD or WITHIN;
   Send RUSH message to all nodes n_i with STATUS[i] = BEHIND;

   Send STOP message to all nodes n_i with STATUS[i] = AHEAD;
   Send RUSH message to all nodes n_i with STATUS[i] = BEHIND or WITHIN;


Else If NODES_AHEAD = = 0  &&  NODES_BEHIND = = 0

   For i:=0 to N-1 do
      Calculate Probability of Divergence P_div^i (equation 5)
      If P_div^i ≤ P_th
         Continue;

      Else
         If the node is ahead of the coordinator
            Send STOP message only to that node;
         Else If the node is behind the coordinator
            Send RUSH message only to that node;
         End If;

      End If;
   End For;

End If:
```

Figure 3: The Movement Algorithm

4. Probability of Divergence

In this paper, we have assumed Random Waypoint Mobility (RWP) model for the calculation of the Divergence Probability of the nodes. In a RWP model, a node randomly chooses a destination point (Waypoint) in the area (a straight line in our case) and moves with constant speed to this point (unidirectional motion). After waiting a certain pause time for sending the POSITION message and receiving the instruction messages, it chooses a new speed, moves with constant speed for the next T/2 interval, and so on. The movement of a node from a starting position to its next destination is denoted as one movement period or transition. The waypoints are uniformly and randomly distributed on the system area.

Let us consider a topological configuration of the network as shown in figure 4. In the topology shown in the figure 4, there are four nodes along with the coordinator within the R_{th} range or the Zone of Stability. Nodes n_A and n_B are ahead of the coordinator while the nodes n_C and n_D are behind the coordinator. The calculation of the Divergence Probability depends on the distance of the nodes from the R_{th} range. This distance l is measured along the direction of motion for each node. Let us call it the Divergence Length.

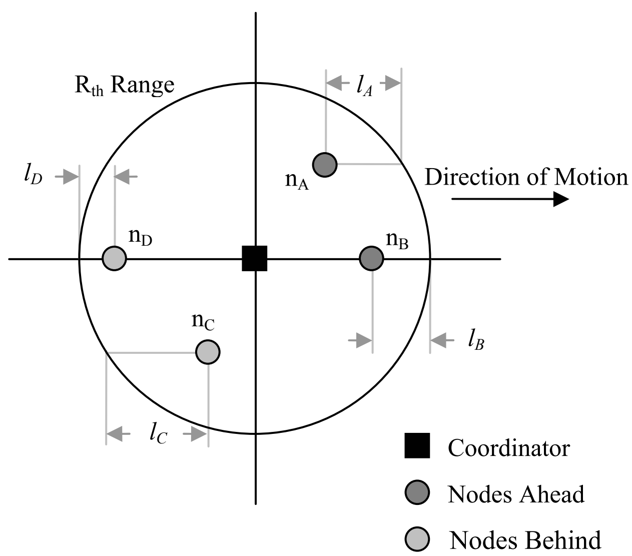


Figure 4: A sample topological configuration of the network

4.1. Calculation of Divergence Length

Let the positional coordinates of the node n_A as obtained from its POSITION message are (x_A, y_A, z_A). Let the coordinates of the coordinator be (x_0, y_0, z_0). For simplicity, we consider only planar motion of the nodes (in the x-y plane) and take the direction of motion along the X-axis. Hence, the distance l_A of node A is given by:

$$l_A = AB = PQ = OQ - OP$$

Again, $OB^2 = OQ^2 + BQ^2$, by virtue of Pythagoras Theorem ($\because BQ \perp OR$). By using the positional coordinates, $BQ = y_A - y_0$, $OP = x_A - x_0$, $OB = R_{th}$

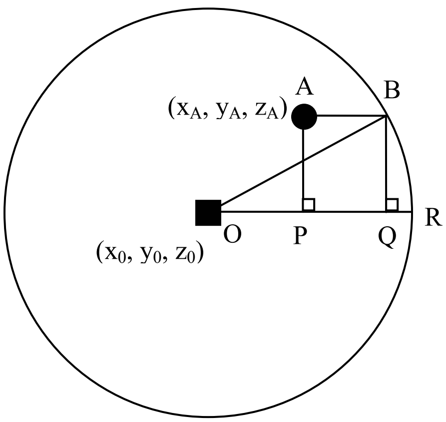


Figure 5: Calculation of Divergence Length

$$\therefore OQ = \sqrt{OB^2 - BQ^2} = \sqrt{R_{th}^2 - (y_A - y_0)^2}$$

$$\Rightarrow l_A = OQ - OP = \sqrt{R_{th}^2 - (y_A - y_0)^2} - (x_A - x_0)$$

, $x_A > x_0$ and $y_A > y_0$

Similarly, for the node n_C, as shown in figure 4, the distance can be obtained as:

$$l_C = \sqrt{R_{th}^2 - (y_0 - y_C)^2} - (x_0 - x_C), \quad x_C < x_0$$

and $y_C < y_0$

Hence, for any node n_i with positional coordinates (x_i, y_i, z_i), the Divergence Length is given by:

$$\boxed{l_i = \sqrt{R_{th}^2 - \left|(y_i - y_0)\right|^2} - \left|(x_i - x_0)\right|} \qquad (1)$$

Henceforth, we will consider only the direction of motion axis for probability calculations and may not consider the Y-axis at all without any loss of generality (as there is no Y-component of node velocities).

4.2. Case 1: The Node is ahead of the Coordinator

Let, in the T/2 interval from nT/2 to (n+1)T/2 (where n is a natural number), the Coordinator has traversed a distance of x while the Node has crossed y length along the Direction of Motion, as shown in figure 6. If the distance between the coordinator and the node was p at t = nT/2, then at t = (n+1)T/2, the distance would be $p+d$, where $d = y-x$ (assuming that the node is ahead of the coordinator and it has got a velocity greater than that of the coordinator in this interval, i.e, $y > x$).

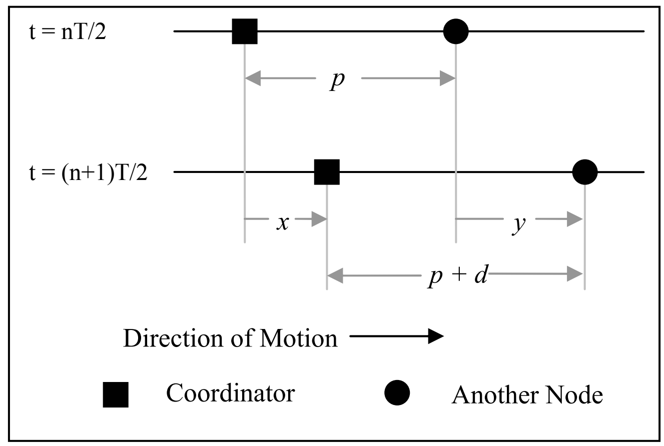


Figure 6: Divergence when a node is ahead

Now, let us assume that at t = nT/2, the node was at a distance l from the R_{th} range (measured along the direction of motion, as per equation 1). Then the node will diverge out of the Zone of Stability at t = (n+1)T/2 if and only if $d > l$. Hence, the Divergence Probability of the node at t = nT/2 is given by:

$$P_{div} = P(d > l) = P(y - x > l) = P(y > x + l)$$

Here, x and y are the distances traversed by the nodes in time T/2, with randomly chosen velocities from within a range [0, V_{max}]. As the velocities are randomly chosen, we can take the distances x and y as random variables having a range of variation [0, a], where a = V_{max}T/2 = X/2. Again, owing to the fact that all the velocities within the range [0, V_{max}] are equally probable, we get the distance variables to be Uniformly Distributed Random Variables over the range [0, a]. So, the Probability Density Function of the distance variables x and y are given by:

$$p_x = \begin{cases} 1/a & \text{for } 0 \le x \le a \\ 0 & \text{otherwise} \end{cases}$$

$$p_y = \begin{cases} 1/a & \text{for } 0 \le y \le a \\ 0 & \text{otherwise} \end{cases}$$

Again, as the coordinator and the node choose their velocities independently in case of Normal Movement, the two Random Variables x and y are independently and identically distributed (i.i.d.). Hence, the joint PDF of these two is given by:

$$p_{xy} = p_x p_y = \begin{cases} 1/a^2 & for\ 0 \le x, y \le a \\ 0 & otherwise \end{cases} \quad (2)$$

Thus, we obtain the expression for Divergence Probability as:

$$P_{div} = P(y > x + l) = \int_0^a \int_{x+l}^a p_{xy}\, dydx = \int_0^{a-l} \int_{x+l}^a p_{xy}\, dydx$$

Now, if $a\text{-}l \le x \le a$, then $a \le y \le a\text{+}l$. Hence, $p_{xy} = 0$. This reduces down P_{div} to:

$$P_{div} = \int_0^{a-l} \int_{x+l}^a p_{xy}\, dydx = \int_0^{a-l} \int_{x+l}^a \frac{1}{a^2} dydx = \frac{1}{a^2} \int_0^{a-l} \left[\int_{x+l}^a dy \right] dx$$

$$= \frac{1}{a^2} \int_0^{a-l} (a - x - l)\, dx = \frac{1}{a^2} \left\{ \left[(a-l)x \right]_0^{a-l} - \left[x^2/2 \right]_0^{a-l} \right\}$$

$$= \frac{1}{a^2} \left\{ (a-l)^2 - (a-l)^2/2 \right\}$$

$$\therefore\ P_{div} = \frac{(a-l)^2}{2a^2} = \frac{1}{2} - \frac{l}{a} + \frac{l^2}{2a^2} \quad (3)$$

4.3. Case 2: The Node is behind the Coordinator

Assuming that the node is behind the coordinator (figure 7) and it has got a velocity less than that of the coordinator in this interval, i.e, $x > y$, we get, at t = (n+1)T/2, the distance between the coordinator and the node as $p+d$ where $d = x\text{-}y$.

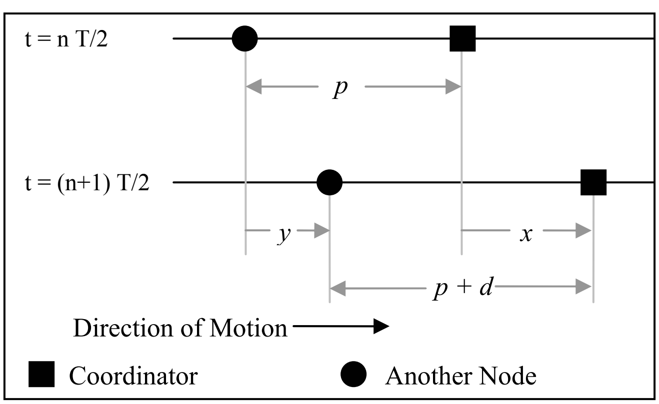


Figure 7: Divergence when a node is behind

Hence, in this case, the Divergence Probability of the node at t = nT/2 is given by:

$$P_{div} = P(d > l) = P(x - y > l) = P(y < x - l)$$

Thus, we obtain the expression for Divergence Probability as:

$$P_{div} = P(y < x - l) = \int_0^a \int_0^{x-l} p_{xy}\, dydx$$

$$= \int_0^l \int_0^{x-l} p_{xy}\, dydx + \int_l^a \int_0^{x-l} p_{xy}\, dydx$$

Now, if $0 \le x \le l$, then $-l \le y \le 0$. Hence, $p_{xy} = 0$. This reduces down P_{div} to:

$$P_{div} = \int_l^a \int_0^{x-l} p_{xy}\, dydx = \int_l^a \int_0^{x-l} \frac{1}{a^2} dydx$$

$$= \frac{1}{a^2} \int_l^a \left[\int_0^{x-l} dy \right] dx = \frac{1}{a^2} \int_l^a (x - l)\, dx$$

$$= \frac{1}{a^2} \left\{ \left[x^2/2 \right]_l^a - \left[l\, x \right]_l^a \right\}$$

$$= \frac{1}{a^2} \left\{ \left(a^2/2 - l^2/2 \right) - l(a - l) \right\}$$

$$\therefore\ P_{div} = \frac{(a^2/2 - la + l^2/2)}{a^2} = \frac{1}{2} - \frac{l}{a} + \frac{l^2}{2a^2} \quad (4)$$

Hence, for any node n_i, the Probability of Divergence is given by:

$$\boxed{P_{div}^{\ i} = \frac{1}{2} - \frac{l_i}{a} + \frac{l_i^{\,2}}{2a^2}} \quad (5)$$

where $a = V_{\max} T/2 = X/2$ and l_i is given by the equation 1.

5. Lemmas

5.1. Lemma 1: Min-Range Selection

The radius of the minimum range of communication should be less than or equal to half of that of the longest communication range for maintaining neighborhood criterion.

Proof: Two nodes n_i and n_j are neighbor to each other if $D(i,j) \le R_{max}$, where $D(i,j) = \| p_i - p_j \| = D(j,i)$. Now, let us consider the position of the coordinator to be p_c. Then, as per our algorithm, we require $D(i,j) \leq R_{max}$, $D(i,c) \leq R_{min}$ and $D(j,c) \leq R_{min}$ for maintaining the network topology.

Now, by triangle law, we get: $|| p_i - p_j || \leq || p_i - p_c || + || p_c - p_j ||$. That is, $D(i,j) \leq D(i,c) + D(j,c) \leq R_{min} + R_{min}$. Therefore, in the worst case, $D(i,j) = 2R_{min}$. Again, we require $D(i,j) \leq R_{max}$. Therefore, we obtain: $2R_{min} \leq R_{max} \equiv R_{min} \leq R_{max}/2$.

Hence, we take the maximum possible limit of the mid communication range, that is $R_{min} = R_{max}/2$ in case of our algorithm.

5.2. Lemma 2: Achieving Convergence

In case of any node going outside the threshold range with respect to the coordinator, this proposed algorithm makes it converge in the zone in no more than a time interval of T.

Proof: Since we are dealing with maximum time required to converge hence we take only the worst possible case to prove it. In the limiting case let us consider that when the last POSITION message was received the node n_k was just at a distance equal to R_{min}.

Case 1: Let, initially, $D(c,k) = R_{th}$ and n_k be ahead of the coordinator n_c. The relative velocity between the node and the coordinator is $V_{rel} = V_k - V_c$. The node will diverge for $V_k > V_c$, that is, for $V_{max} \geq V_{rel} > 0$. After time interval $T/2$, we get, $D(c,k) = R_{th} + V_{rel}T/2$. Therefore, $R_{th} < D(c,k) \leq R_{th} + V_{max}T/2$. That is, $R_{th} < D(c,k) \leq R_{th} + X/2$ as $V_{max}T = X$. Now, the node n_k is stopped and the other nodes rush for an interval of $T/2$. So, after an interval of $T/2$, we get, $D(c,k)_{current} = D(c,k)_{previous} - V_{max}T/2$. Therefore, $R_{th} - X/2 < D(c,k) \leq R_{th}$ after a total time of $T/2$ from divergence. So, n_k has converged within a time interval of $T_{conv} = T/2$.

Case 2: Let, in this case, the initial distance of n_k from the coordinator n_c be $D(c,k) = R_{th}$, but n_k is behind the coordinator. Here, n_k will diverge for $V_k < V_c$, that is, $0 > V_{rel} \geq -V_{max}$. After time interval $T/2$, we get, $D(c,k) = R_{th} - V_{rel}T/2$. Therefore, $R_{th} < D(c,k) \leq R_{th} + X/2$. Now as n_k is behind the coordinator and $D(c,k) > R_{th}$, all the other nodes are stopped and n_k rushes for an interval of $T/2$. After the interval $T/2$, we obtain, $D(c,k)_{current} = D(c,k)_{previous} - V_{max}T/2$. Therefore, $R_{th} - X/2 < D(c,k) \leq R_{th}$. So, n_k is converged within a time interval of $T_{conv} = T/2$.

Case 3: Let, in this case, two nodes get diverged form the threshold zone – one ahead of the coordinator and the other behind it. In such a case, First, all the nodes within the range R_{th} and the node diverged ahead are asked to stop while the node diverged behind is instructed to R_{th} for a time interval of $T/2$. After $T/2$, the node behind gets converged (as proved in Case 2). After this, the node that had diverged ahead is instructed to stop while the coordinator rushes along with the well-connected nodes for a time interval of $T/2$, which in turn converges the node ahead (as proved in Case1). Thus, both the diverged nodes have been converged back into the threshold zone within a total time interval of $T_{conv} = T/2 + T/2 = T$.

As we have considered the worst possible situations on both the ends, we can state that the total time required for the convergence of a node is always less than or equal to T. That is, $T_{conv} \leq T$. This proves the lemma.

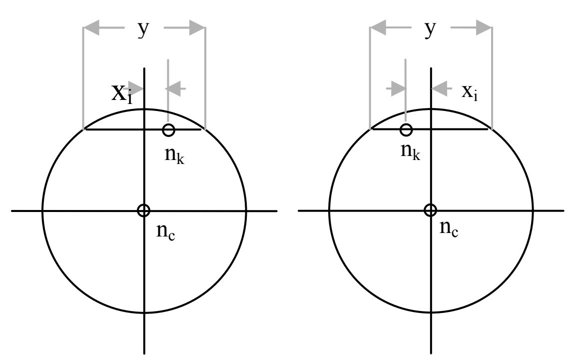
Lemma 3: Stable Zone

If a node is within the zone of stability as defined before, it will either be stable or can be made stable easily after divergence.

Proof: We take a node n_k within the circle R_{th} such as the horizontal band on which it resides has a length of y, as shown in figures 8. Let the initial horizontal distance of the node n_k from the coordinator n_c is given by x_i.

Case 1: Let n_k be *ahead* of n_c, as in figure 8(a). So, if the relative velocity $V_{rel} = V_k - V_c$ be positive, then after a time $T/2$, the distance is $\boldsymbol{D(c,k) = x_i + V_{rel}T/2 = x_i + X_{rel}/2}$, where we denote $X_{rel} = V_{rel}T$. Now, if $D(c,k) = x_i + X_{rel}/4 > y/2$, the node n_k diverges and so it is stopped while the other nodes rush for a time $T/2$. After this interval of $T/2$, we get: $\boldsymbol{D(c,k) = x_i + X_{rel}/2 - V_{max}T/2 = x_i + X_{rel}/2 - X/2}$

The divergent node n_k will be converged if and only if $| x_i + X_{rel}/2 - X/2 | \leq y/2$. If we assume that in the first interval of $T/2$, n_k was just diverged, that is, $x_i + X_{rel}/2 \approx y/2$, then we obtain: $\boldsymbol{| y/2 - X/2 | \leq y/2 \equiv y \geq X/2}$.



8(a) 8(b)

Figure 8: Stable Zone Lemma

Case 2: On the other hand, let us suppose that n_k was x_i distance <u>behind</u> n_c at the beginning, as in figure 8(b). So, if the relative velocity $V_{rel} = V_k - V_c$ be negative, then after a time $T/2$, the distance is $\boldsymbol{D(c,k) = x_i + V_{rel}T/2 = x_i + X_{rel}/2}$. Now, if $D(c,k) = x_i + X_{rel}/2 > y/2$, the node n_k

diverges and so it is stopped while the other nodes rush for a time $T/2$. After this interval of $T/2$, we get:

$\boldsymbol{D(c,k) = x_i + X_{rel}/2 + V_{max}T/2 = x_i + X_{rel}/2 + X/2}$

Now, the node n_k rushes for a time interval of T while the other nodes remain stationary. After this interval T, we get: $\boldsymbol{D(c,k) = x_i + X_{rel}/2 + X/2 - V_{max}T = x_i + X_{rel}/2 - X/2}$. So, similar to Case 1, the condition that $\boldsymbol{y \geq X/2}$ has to be true so as to make n_k converge within time interval of $3T/2$ after divergence.

As we have considered both the extreme cases of motion for node n_k, we can state that the horizontal band on which a node resides while in motion should have a length of $y \geq X/2$ to satisfy the convergence claim (Lemma 1) of our algorithm. We have defined the ***Zone of Stability*** as the region where $\boldsymbol{y \geq X/2}$. So, any node residing within the zone will surely follow the algorithm as well as the claim of convergence within time $3T/2$.

6. Experimental Results

The proposed algorithm was simulated using C programming language on an MS-DOS platform. The simulation was performed on a synthetically designed situation where a MANET with 5 nodes has been considered. All the nodes of the network are allowed to move with their preferred random velocities unless directed otherwise by the movement coordinator.

6.1. Simulation Result

The initial condition for the simulation is chosen as follows: R_{max} = 100 km, R_{min} = 50 km, R_{th} = 40 km, V_{max} = 60 km/hour and hence $T = (R_{min} - R_{th}) / V_{max}$ =10 min.

The positions of the coordinator were randomly selected by the algorithm and their velocities were also randomly selected within the allowed range.

The results obtained through this simulation are presented in figure 9. The figure shows the plot of the Distance of each Node from the Coordinator (in km) versus Time (in min) over an interval of 10 hours = 600 minutes. The horizontal margin at 40 km marks the Threshold range (R_{th}) and the topmost margin at 50 km marks the Min-Range of Communication (R_{min}). It is seen that within the simulation time, none of the nodes have gone out of the Threshold Range. Also it was seen that the total number of control overheads required during the whole simulation run was only 8. Thus this algorithm maintains the stable network configuration efficiently and at the same time uses very little control overheads.

6.2. Performance Comparison

We have also performed a comparative study of the proposed algorithm with a previous one proposed by us in [8]. The performance comparison is done through the simulation of both the algorithms on a set of synthetically designed network scenarios. In all the simulations, the time intervals are in minutes and the distances are in kilometers. We have considered five nodes and have plotted the distance of four nodes from the coordinator against time. In both cases T has been taken as 10 minutes. The simulation results for the proposed stochastic algorithm are shown in figure 9. The initial conditions are as given in section 6.1. The simulation results for the previous algorithm proposed

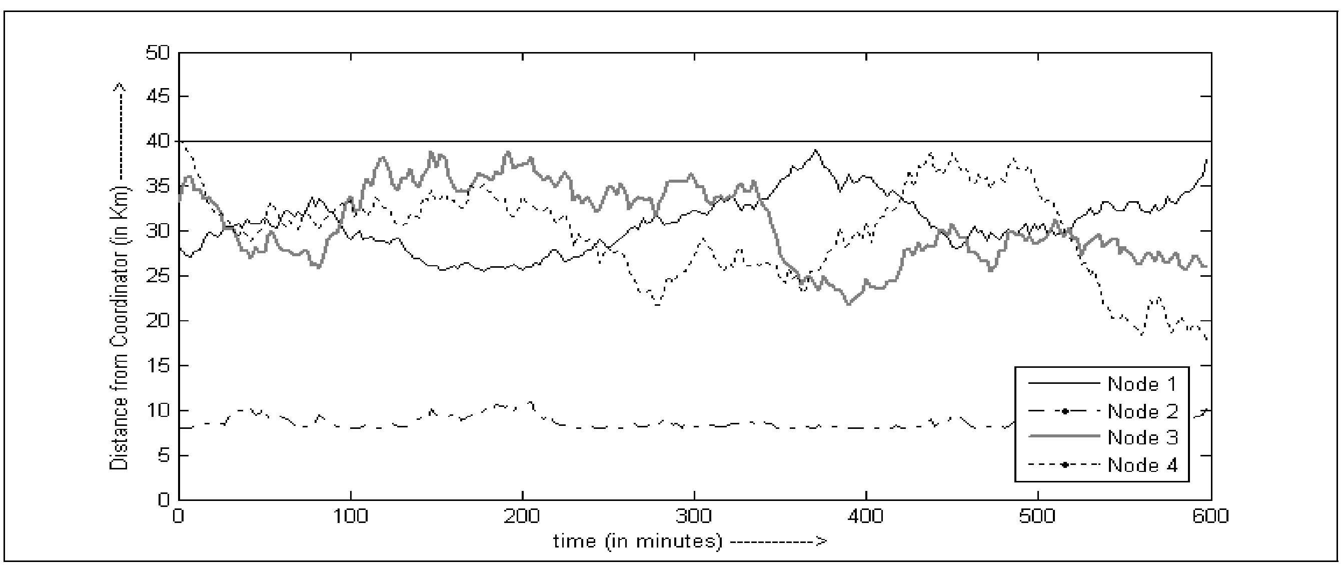


Figure 9: Simulation results of the proposed algorithm

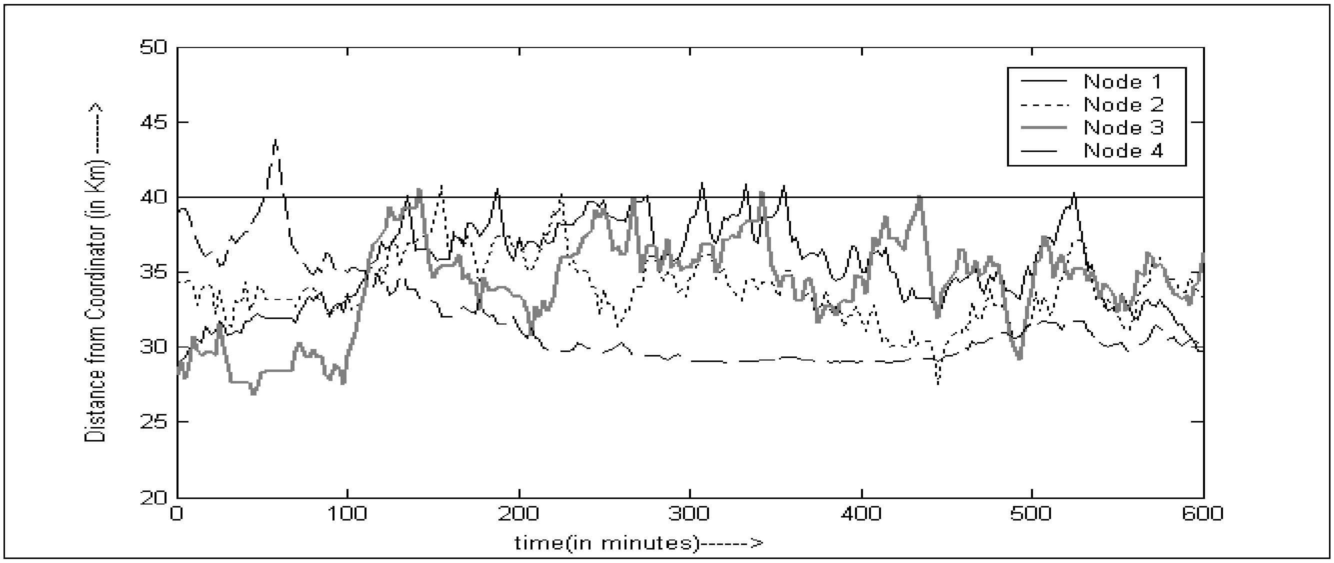


Figure 10: Simulation results of the algorithm proposed in [8]

in [8] are shown in figure 10. For the simulation purpose we have taken R_{max} = 100 km, R_{mid} = 50 km, R_{min} = 40 km, V_{max} = 60 km/hour and hence $T = (R_{mid} - R_{min}) / V_{max}$ =10 min. It is seen that for the previous algorithm of [8], some nodes have diverged out of the 40Km range i.e R_{min} (R_{min} of [8] is equivalent to R_{th} of our proposed algorithm). This has not happened in case of the proposed algorithm. This demonstrates its efficiency in maintaining a stable network topology. Its efficiency is due to the fact the even before a node can diverge, the coordinator will detect its tendency to diverge using the stochastic methods described above and take necessary actions to prevent divergence.

7. Conclusion

In this paper, we have developed a stochastic topology management algorithm. This scheme controls the movements of the nodes of the MANET depending upon the outcomes of certain stochastic calculations and thus retains the network configuration of the MANET. The simulation result as well as the comparison with the algorithm in [8] proves the efficiency of the proposed algorithm. Presently, we are working on a distributive approach of the algorithm which will also allow the nodes to move in random directions.

8. Reference

[1] National Institute of Standards and Technology, "Mobile Ad Hoc Networks (MANETs)", *http://w3.antd.nist.gov/wahn_mahn.shtml*

[2] IEEE 802.11. "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification", *Standard, IEEE, New York, November 1997.*

[3] Elizabeth M. Royer and Chai-Keong Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", *IEEE Personal Communications, Vol. 6, No. 2, pp. 46-55, April 1999.*

[4] M J Ng and I T Lu, "A Peer-to-Peer Zone based Two-Level Link State Routing for Mobile Ad Hoc Networks.", *IEEE Journal on Selected Areas in Communications*, *Special Issue on Ad-Hoc Networks, August 1999, p 1415.*

[5] C K Toh, "Long-lived Ad-Hoc Routing based on the Concept of Associativity", *IETF Draft, March 1999.*

[6] M. Jiang, J. Li and Y. C. Tay, "Cluster Based Routing Protocol (CBRP)", *http://www.ietf.org/internet-drafts/draft-ietfmanet-cbrp-spec-00.txt, August 1998.*

[7] Sourav Sen Gupta, Surjya S. Ray, Somnath Samanta and Mrinal K. Naskar, *"An Efficient Topology Management Algorithm for Mobile Ad-Hoc Networks", International Conference on Electronic and Photonic Materials, Devices and Systems – 2006 (EPMDS 2006).*

[8] Somnath Samanta, Surjya S. Ray, Sourav Sen Gupta and Mrinal K. Naskar, *"A Novel Algorithm for Managing Network Configuration of a Mobile Ad-Hoc Network", 4th Asian International Conference on Mobile Computing - 2006 (AMOC 2006).*