Greedy Cylinder Routing Algorithm for Mobile Network Devices

ALaa E. Abdallah
Faculty of Information Technology
Hashemite University
Zarqa 13115, Jordan
Email:aabdallah@hu.edu.jo

Emad E. Abdallah
Faculty of Information Technology
Hashemite University
Zarqa 13115, Jordan
Email:emad@hu.edu.jo

Abstract—In geographical routing algorithms, mobile nodes rely on geographical position to make routing judgments. Researchers frequently discuss such routing algorithms in (2D) space. But, in reality, mobile nodes spread in (3D) space. In this paper we present two new 3D geographical-based routing algorithms Cylinder and GCylinder. In Cylinder routing, the nodes are locally projected onto the inner surface of a cylinder, face routing is executed after that. Greedy-Cylinder starts with Greedy routing algorithm until a local minimum is reached. The algorithm then switches to Cylinder routing. We evaluate our new algorithms and compare them with greedy algorithm. The simulation outcomes show the substantial enhancement in delivery rate over other classical known algorithms.

I. INTRODUCTION

Mobile ad-hoc network (MANET) is an infrastructureless network which consists of a group of wireless mobile nodes that can communicate with each other wirelessly. A mobile node \( x \) in MANET can communicate directly only with the nodes inside a virtual sphere centered at \( x \) of radius \( R \) (the transmission range). To communicate with nodes outside its transmission range, multihop routing is used utilizing intermediate communicating nodes. Since mobile ad-hoc networks may change their topology frequently and because of the resource constraints, routing in such networks is difficult. Numerous routing algorithms for MANET have been proposed to address the multi-hop routing problem.

In general, Ad hoc routing algorithms can be classified in two basic types: static routing and online routing. Static routing algorithms use a predefined route among the network hosts using the information about the virtual links in the network [2], [5], [11], [16]. Online routing algorithms use the nodes (source, destination, and neighbors) geometric location to forward the packets in the direction of the destination [15], [17], [1], [13], [6], [9], [4], [3]. In Greedy routing [6], a node forwards the packet to one of its neighbors that minimize the Euclidean distance to the destination. Directional routing [13], the current node forwards the packet to one of the neighbors that minimize the angle between current node, next node, and destination.

Clearly, Greedy and directional routing fail to deliver the packet at local minimum (when there is no way to progress any more). The problem has been solved in 2D, using what is called face routing [4], [12], [7], [3]. Face routing can be explained as follows: (a) nodes locally use their geometric locations to extract a planar sub-graph i.e. Gabriel graph (GG)[8] or Relative Neighborhood Graph (RNG)[10], (b) Face routing traverses in a counter clockwise the faces of GG that cross the virtual line between the source and the destination. (c) The algorithm switches between faces, when the packet reaches an edge that crosses the virtual line between the source and the destination at a point closer to the destination than any previously known intersection point. It has been proved that face routing has a guaranteed packet delivery.

Many of online routing algorithms are widely used in 2D MANETs, but their performance on MANETs embedded in 3D space is not good [18]. In this paper we propose two online routing algorithms. Cylinder routing, in this algorithm the nodes are locally projected onto the inner surface of a cylinder and then face routing is executed. The second algorithm is called GCylinder, it starts with Greedy routing algorithm until a local minimum is reached. The algorithm then switches to Cylinder routing.

The rest of the paper is organized as follows: In Section II, we briefly present the network model. A cylinder projection algorithm needed in our protocols is described in the same section. In Section III, we introduce our local routing algorithms. In Section IV we present experimental results to demonstrate the much improved performance of the proposed method in comparison with existing techniques. We conclude our paper in Section V.

II. PRELIMINARIES

A. The Communication Network Model

In the following we define the considered network model. Consider a set of wireless mobile hosts in 3D space. The transmission range of each network host is represented as a sphere volume of radius \( R \), where \( R \) is the maximum transmission range. A unit ball graph (UBG) is widely used to represent the network. In UBG, two nodes \((u, v)\) are connected by an edge if \( u \) is inside the sphere centered at \( v \). Given a UBG, the problem of online routing is to find a path in UBG from any node to any other node. At each host of the path, the choice of the next host is based on the local information of the geometric locations of the source and the destination.

Evaluating the new algorithms are done through the delivery rate (the percentage of times that each algorithm succeeds in delivering its packet) and the path dilation (the average ratio of the length of the path returned by the algorithm to the length of the shortest path in the UBG).
B. Cylinder projection

Here, we define our algorithm to project 3D UBG on the inner surface of a cylinder. The algorithm starts by translating all nodes to make the source node \( s \) on the origin (This can be done locally). Then a rotation around \( x \) axis followed by a rotation about \( y \) axis is done for all nodes as well. After this step, the source position will be located at the location \((0, 0, 0)\) and the destination node will be located at \( z \) axis \((0, 0, z)\). See Figure 1. After translation, the nodes are projected onto the surface of a cylinder as follows, see Figure 2:

1) The \( z \) axis of the point will be the height of the projected point on the inner surface of the cylinder.
2) The angle between the \( x \)-axis of the point and the \( y \)-axis of the point gives the length of the curve on the projected surface.
3) The source and destination are projected onto the \( x \)-axis of the cylinder.

III. Proposed Routing Algorithms

In the following we assume that the current node is \( c \), the source node is \( s \) and the destination node is \( d \).

A. Cylinder Routing

From the cylinder projection algorithm described in section II-B, it is clear that using a constant number of arithmetic operations each host can define locally its’ own new 2D position, its neighbors new positions, and the new position of the destination node. Face routing is performed after that. Algorithm 1 gives a description of the Cylinder routing algorithm.

Projection cylinder routing improved the delivery rates but the path dilation is increased dramatically. See Figure 3 and Figure 4. In the following, we define our solution for the path dilation increase.

B. GCylinder routing algorithm

Our local algorithm to solve the problem of the high path dilation from previous algorithm is to use the advantage of greedy routing, it starts by using greedy routing until the progress is impossible (reach local minimum). The algorithm then switches to Cylinder routing. Algorithm 2 gives a description of the GCylinder routing algorithm.

IV. Simulation Results

In this section we describe our simulation environment, demonstrate and interpret the results, and compare the new algorithms with previously published online routing algorithms.

A. Simulation environment

In the simulation experiments, 100 nodes are randomly generated in a cube of side length 100. The maximum transmission radius of each host is set to 25. We first calculate all connected components in the graph. Then select the largest connected component among all the connected components to perform the routing algorithms. The source and destination nodes are then randomly picked from the largest connected component. It is suggested in [14] to consider simulations with node density per unit disk of around 5 in 2D environment, which would correspond to the graph with average node degrees of around 4. To compute the packet delivery rate, this process is repeated with 100 random graphs and the percentage of successful deliveries determined. To compute the average packet delivery rate, the packet delivery rate is determined 100 times and an average taken. Additionally, out of the 10000 runs used to compute the average packet delivery rate, the path dilation is computed.
Algorithm 1: Cylinder Routing

// Algorithm is executed independently by the current node.
// Execution starts either when a source node starts to route a packet to some other node, or if the node \( c \) receives a packet from a neighbor nodes and need to route it.

begin
1. \( c \) determines its 2D position using the cylinder projection algorithm in section II.
2. \( c \) finds all its neighbors position and obtains their 2D coordinates.
3. \( c \) determines the 2D position of the destination node using cylinder projection algorithm.
4. \( c \) applies 2D face routing using the new coordinates to determine which neighbor it has to forward the packet.
end

Algorithm 2: GCylinder Routing

// Algorithm is executed independently by the current node.
// Execution starts either when a source node starts to route a packet to some other node, or if the node \( c \) receives a packet from a neighbor nodes and need to route it.
// Let \( N(c) \) is the set of neighbors of \( c \).

begin
1. The current node \( c \) forwards the packet to the neighbor node \( u \in N(c) \) that minimizes the remaining distance to the destination node \( d \).
2. If there is no neighbor closer to the destination than the current node \( c \) determines its 2D position using the cylinder projection algorithm in section II.
3. \( c \) finds all its neighbors position and obtains their 2D coordinates.
4. \( c \) determines the 2D position of the destination node using cylinder projection algorithm.
5. \( c \) applies 2D face routing using the new coordinates to determine which neighbor it has to forward the packet.
end

B. Observed Result

We present a comparison between different algorithms in terms of packet delivery rate and path dilation in Figure 3 and Figure 4. It is immediately evident form the result given in Figure 3 that Greedy and Compass have the lowest delivery rate less than 65% which yields the low path dilation because the packets that fail to arrive to the destination is not counted in the path dilation. The delivery rate of our new algorithm Cylinder routing jumps to 80%, but this algorithm has by far the worst path dilation (around 8), the best delivery rate with over 95% is found in our second algorithm GCylinder, and also has a lower path dilation than Cylinder routing (around 3.87).

V. Conclusions

In this paper we propose two new online routing algorithms for routing in 3D ad-hoc networks, based on the idea of projecting 3D nodes onto the inner surface of cylinder. Simulation results demonstrate that our new algorithms (Cylinder routing and GCylinder routing) yield a definite improvement over greedy algorithm in terms of delivery rate and path dilation.
Fig. 4. Path Dilation for the proposed algorithms

REFERENCES


