© November 2015 | IJIRT | Volume 2 Issue 6 | ISSN: 2349-6002

Motion Vector Routing in Ad-Hoc Transport Networks

RAKESH KUMAR GIRI

Research Scholar, Department of Computer Science & Engineering, Sunrise University, Alwar, Rajasthan

Abstract— Vehicular networks are an important application of Mobile Ad-hoc Network (MANET) technology. Motion-vector routing attempts to exploit the inherent structure of node motion within a road system. Simulation results demonstrate the effectiveness of this technique in reducing network topology changes between nodes moving in the same direction. The work performed is independent of any specific MANET routing protocol. Further work is needed to quantify the expected improvement when operating in conjunction with existing routing protocols.

Index Terms- MANET, MV ROUTING, SIMULATION, GPS

I. INTRODUCTION

Mobile Ad-hoc Networks (MANETs) are finding applications in an increasing number of unconventional areas. One such new application is vehicular networking, in which vehicles traveling on a road system comprise the nodes of the network. Many uses exist to motivate the development of vehicular networks, including distributed traffic statistic generation and collection, safety hazard warning dissemination, and general purpose entertainment and information services.

While most of these services could be provided via infrastructure networks, such networks suffer from a high deployment cost as well as an all-or-nothing limitation where applications may not be saleable unless near 100% availability is achieved. A hybrid infrastructure/MANET approach in which the mobile nodes form a "glue" MANET tying together widely distributed infrastructure points could reduce both the total cost of operation and deployment costs of a network[1].

The bulk of work to date on MANETs have evaluated new protocols and algorithms with highly general node distribution and motion models. A common example is the random motion model used in the simulation of many proposed protocols. The motion observed in vehicular network nodes, however, is constrained to the road system. It is consequently possible to exploit these properties by slightly modifying the neighbor selection algorithm to consider the relative motion between two candidate nodes when choosing neighbors. This technique attempts to improve the stability of the network's logical topology, and consequently the overall efficiency and stability of the network.

II. RELATED WORK

Prior work on this general class of problem involved one of two approaches. *Geocasting* addresses each message to a particular geographical area rather than a unique node.[2] The network is responsible for delivering the message to all nodes with the specified area. Networks of this type are sometimes referred to as using *data-centric* addressing.

A similar but distinct approach is location-based routing, in which the each node chooses routes for messages based on the absolute position of neighbors relative to the absolute position of the receiving node.[3] Both location-based approaches have merit but share common disadvantages when applied to vehicular networking. The use of absolute positioning information for routing requires GPS or an equivalent system which increases node cost and power requirements. At the same time, robustness decreases as satellite-based positioning systems do not provide 100% availability. Absolute position calculations can also be processor-intensive, imposing increased CPU and power overhead at the network layer. Finally, position information requires trend analysis to equal the predictive properties of a velocity vector – why not directly measure velocity instead?

III. PROPOSED APPROACH

At a microscopic scale, vehicles traveling on a road system could be generalized as nodes traveling bi-directionally on a line. This model fails to hold at the macroscopic level, but given the limited range of radios that might be used for such a system (802.11b, etc) the model is relevant to all but the most urban road layouts. The inclusion of a motion vector in neighbor advertisements would allow a node to quickly categorize potential neighbors into two groups: nodes moving *with* or *against* the node in question. This would allow a node to eliminate undesirable neighbors. A less severe implementation would at least prioritize nodes with less relative motion over those with more.

The work herein assumes that a useful network quality metric is the network's topology stability – defined here as the average time that any two nodes are neighbors (that is, a node's neighbor selection algorithm admits a given other node). Further work will be necessary to fully quantify the correlation between this metric and the actual performance of existing and proposed ad-hoc routing protocols running with the motion-vector modification.

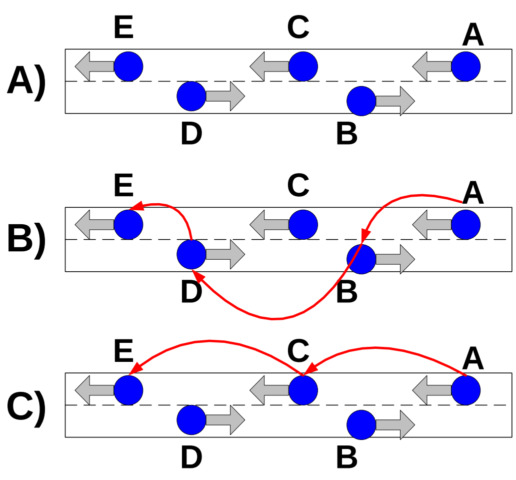

Figure 1. A) Near-scale model of vehicular network. B) Non-optimal routing. C) Optimal routing

An important point to make is that this proposed neighbor selection strategy may only turn out to be useful for applications that inherently limit interactions to vehicles traveling in the same direction. It is reasonable for motion-vector routing to improve network topology stability between two endpoints moving in the same direction. Endpoints moving in opposite directions, however, present a fundamental problem of changing topology that cannot be overcome by selecting a certain path.

Implementation of the motion-vector routing technique as a retrofit to an existing ad-hoc routing protocol would require the addition of motion information to neighbor broadcasts. The relative motion heuristic would be added to the neighbor selection process either as a filter or as a weighted consideration. Implementations could choose to weight neighbor choices according to relative motion, or a hard limit on maximum relative motion could be imposed. Care must be taken to avoid completely partitioning the two directions of travel in cases where a fully routable general-purpose network is desired.
Simulation

IV. METHODOLOGY

A time-stepped Java simulation was developed to further explore the concept of motion-vector routing. The domain object model allows for the simulation of a 2-dimensional space with any number of node objects. Each node points to pluggable motion and neighbor-selection strategy objects. Exposed as Java interfaces, the APIs for motion and neighbor-selection allow for the simulation of any arbitrary combination of motion and neighbor-selection algorithms.

To limit the complexity of the simulation, no MANET routing protocol was implemented following the assumption that network topology stability – as represented by the "average neighbor duration" described above – is a reasonable metric for network quality. The simulation is architected to enable the effect of different combinations of motion and neighbor selection strategies on the average neighbor duration interval.

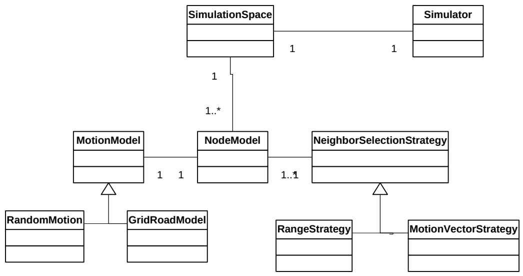

Figure 2. UML representation of simulation domain objects

For the purposes of this paper, two motion models were implemented. The first is a random motion model intended to resemble those used in simulations of other proposed MANET protocols and algorithms. The intent of the random motion model is to generate baseline data to allow the comparison of motion-vector routing with other works to date. The random motion model is parameterized with a maximum velocity (in distance units/time step), and maximum state time (in time steps). Figure 3 details the precise behavior of the random motion model.

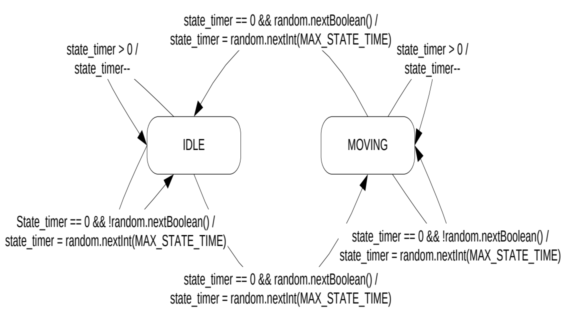

Figure 3. State diagram for random motion model

The second motion model is a grid road model designed to resemble an urban grid road network. Nodes are assigned a turn probability along with minimum and maximum speeds. The model is given a set of column and row road coordinates. For the purpose of this simulation, roads are spaced evenly every 40 units in both the horizontal and vertical dimensions. At initialization, nodes are placed randomly on one of the roads and assigned a random direction and speed. As time increases the node's position is incremented at the prescribed velocity. Every time an intersection is passed, the decision to turn is made with the assigned probability. If a turn is made, the direction (right or left) is random and a new random speed is assigned.

Two neighbor selection strategies were implemented for this paper. One is a radio range-based strategy that accepts all nodes within a specified radius. The second is the motion vector strategy described in the section above simplified to take a single parameter in addition to absolute radio range. The motion vector strategy used for simulation will accept any node whose motion is within a specified angle of the local node's motion, as well as any node that is not moving. The logic behind this strategy is diagrammed in Figure 4.

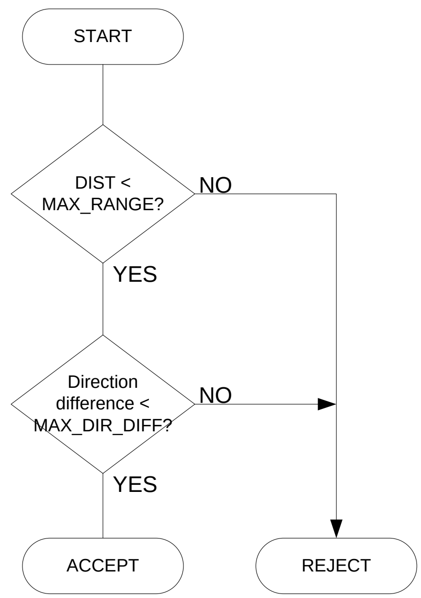

Figure 4. Operation of motion vector neighbor selection strategy

V. RESULTS

Simulations were run combining grid and road motion models with both range and motion vector neighbor selection strategies. Simulation parameters for both motion models are shown below in Table 1. For both motion models, the motion-vector strategy was run with 3 different angle tolerances: 60, 90, and 120 degrees. Due to significant deviations between successive runs, every simulation configuration presented herein was run 6 times with the individual results and overall average represented on a scatter plot. Simulation iterations were hand-tuned to strike a balance between reasonable run time and convergence on a single value. The number of nodes in each motion model was chosen to maintain a steady node density. This was done so that numbers generated for both motion models would be roughly comparable.

Parameter	Random	Grid Road
X dim	1000	1000
Y dim	1000	1000
Number of nodes	400	16
Max state time	10	N/A
max velocity	3	3
min velocity	N/A	1
turn probability	N/A	0.1
max radio range	30	30
sim iterations	400	20000

Table 1. Simulation parameters

Figure 5 summarizes the performance of both range-based and the proposed motion-vector strategies with a random motion model. A significant reduction in neighbor association duration is observed when using the motion vector strategy over a simple range-based strategy. This is in line with expectations since nodes following a random motion model have a highly chaotic path. Our expectation is that performance of the motion vector strategy would improve if the random model were biased towards longer state times. Doing so would increase the average length of time that each node spends moving in a given direction, making the act of taking direction into account more powerful. These results indicate that the motion-vector strategy would not perform well with networks in which the nodes truly mode randomly.

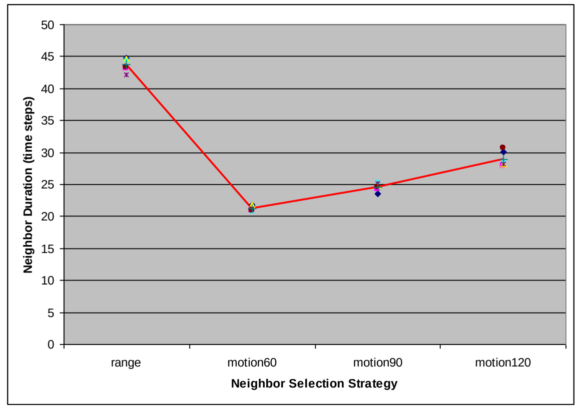

Figure 5. Performance comparison between random and motion vector strategies with a random motion model.

Node motion is significantly less chaotic in the grid road model – especially with if the model is configured with a low turn probability. The simulation results in Figure 6 confirm that the motion-vector strategy is superior to range-based neighbor selection with nodes moving according to this model. The performance of the motion-vector strategy with maximum angle differences of 60° and 90° was nearly identical, improving the average neighbor adjacency interval by nearly a factor of 2. The motion vector strategy with a maximum angle difference of 120°, however, performed poorly. The performance gap between the 120° strategy and the others is understandable given that all traffic in the grid motion model runs at 90° angles. Consequently, the 60° and 90° models exclude perpendicular traffic while the 120° model would include it.

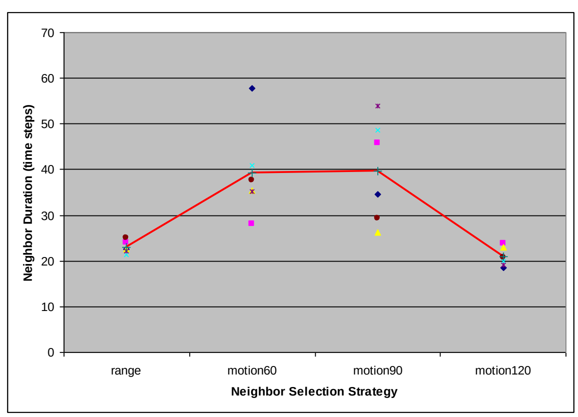

Figure 6. Performance comparison between random and motion vector strategies with a grid road motion model.

CONCLUSION

The motion vector routing technique has the potential to improve network topology stability for networks with a structured and predicable motion model such as the grid road motion model described herein. Networks with less structured motion are not, however, good candidates for motion-vector routing. While motion-vector routing demonstrated a nearly 2X improvement in in topology stability when used on a grid road model, the same strategy was significantly worse than the baseline when used with a random motion model.

More work is necessary to correlate our measure of network topology stability with the real-world performance of leading ad-hoc routing protocols. More analysis must also be performed on the requirement of maintaining connectivity to nodes moving in the opposite direction and the methods for accomplishing this.

REFERENCES

[1] Dousse, Olivier, et al. ‘Connectivity in ad-hoc and hybrid networks’.

[2] Schwingenschlögl, Christian and Kosch, Timo. ‘Geocast Enhancements of AODV for Vehicular Networks.’ Mobile Computing and Communications Review. Vol 6, Number 3.

[3] Lin, Xu and Stojmenovic, Ivan. ‘GPS based distributed routing algorithms for wireless networks.’