

Analysis of a Geolocation-Assisted Routing Protocol for Airborne Telemetry Networks

Kevin Peters, Egemen K. Çetinkaya, Abdul Jabbar, and James P.G. Sterbenz

Department of Electrical Engineering & Computer Science

The University of Kansas

Lawrence, KS 66045

{kevjay, ekc, jabbar, jpgs}@ittc.ku.edu

ABSTRACT

Emerging networked telemetry systems require domain-specific routing protocols, such as AeroRP, to cope with the challenges faced by the aeronautical environment. We present an ns-3 based performance analysis of the geolocation-based forwarding and store-and-haul mechanisms used by AeroRP. The analysis of the simulations shows AeroRP has several advantages over other MANET routing protocols and offers tradeoffs for different performance metrics in the form of different AeroRP modes.

I. INTRODUCTION AND MOTIVATION

Modern telemetry systems require multihop transmission of the data for a networked telemetry system and this need is recognised by the iNET community [1]. However, the highly dynamic environment poses unique challenges such as short transmission times between test articles due to speed and limited connectivity due to mobility. Therefore, a domain specific geolocation-based routing protocol, AeroRP, is proposed for multihop routing in networked telemetry systems [2]. The main focus of AeroRP is to efficiently route data packets, specifically telemetry data, amongst airborne nodes (ANs) to a ground station (GS). There can also be specialized nodes, relay nodes (RN), whose sole purpose is to facilitate the routing of data by the ANs. The ANs must use themselves or RNs as next hops in order for the packets to reach their destination as the AN may never be within transmission range of the GS within a reasonable amount of time.

Mobile Adhoc Networks (MANETs) are self configuring wireless networks with no pre-established infrastructure. Routing packets amongst a network in which a specific hop-by-hop path will most likely not persist must be a major consideration by the MANET routing protocol since ANs can have relative speeds up to Mach 7. These fast moving nodes create a unique challenge for routing packets when connectivity amongst the nodes is very intermittent and episodic. Thus, traditional MANET routing protocols are not suitable for such environments. On the other hand, the different geographic-based routing protocols that are available do not seem to take high velocity of the nodes into major consideration. However, when traveling at such high speeds, the velocity of the node with respect to the destination can be an important consideration for routing. This trajectory data is also important for predicting nodes that will not be within transmission range when nodes are moving very quickly in and out of transmission range of one another.

In this paper, we first present a brief overview of the AeroRP protocol and calculations required to make decision to forward the packets to the next best available hop. Furthermore, a heuristic metric that takes transmission range and a node's location and velocity into consideration is detailed. Next, we present an implementation of AeroRP in ns-3. We also present *preliminary* performance results of the geographic-based AeroRP routing protocol and compare the performance of AeroRP to legacy MANET protocols. The scenarios explored in this research are bound by the highly dynamic and airborne iNET use cases [1, 3, 4, 5]. We show that certain modes of AeroRP outperforms the MANET routing protocols in terms of succesful packet delivery in this highly-dynamic environment.

The rest of the paper is organized as follows: the next section discusses background information specific to geographic routing protocols and routing data in high speed networks. The AeroRP decision metrics are detailed and AeroRP flow is presented. Then, we present the simulation results comparing AeroRP to traditional MANET protocols in a highly dynamic network. Finally, conclusions and future work is presented.

II. BACKGROUND AND RELATED WORK

We present some existing multihop, geographic routing strategies in this section. The various geographic routing survey papers [6, 7, 8] break down different geographic forwarding decisions into MFR (most forward with radius r), NFP (nearest with forward progress), and compass. MFR is the most intuitive and forwards the packet to the node, which makes the most forward progress between the source and destination. NFP forwards the packet that is closest to the current node and is closer to the destination, reducing packet collisions compared to MFR by making shorter hop routing decisions. Compass forwarding chooses a node that is closest to an imaginary line drawn between itself and the destination based on the trajectory. There are many popular geographic routing protocols, including DREAM [9], LAR [10], GPSR [11], and SiFT [12].

Unpiloted aerial vehicle geographic routing has been simulated at 25 m/s [13], much slower than Mach 3.5 (1200 m/s). A top speed of 20-50 m/s is typical amongst geographic routing protocols [14, 15, 16, 17, 18, 19, 20]. There are a few routing protocols specifically for aeronautical environments. ARPAM [21] is a hybrid AODV [22] protocol for commercial aviation networks that utilizes the geographic locations to discover the shortest but complete end-to-end path between source and destination. multipath Doplar routing (MUDOR) [23] takes relative velocity into consideration as well as the Doppler shift to measure the quality of a link. Anticipatory routing [24] tracks highly mobile endpoints that reach the reactive limit in which the speed of the nodes is comparable to the time it takes for the location tracking to converge upon the position of the node. Spray routing [25] involves unicasting a packet a specific *depth* away from the destination in which the packet is then *sprayed* or multicasted to a controlled *width* or number of levels of neighbors, for highly mobile endpoints up to 250 m/s. However, none of these approaches mention such speeds as high as Mach 3.5 in which rapidly varying connectivity is a major consideration.

III. AeroRP DECISION METRICS

The basic operation of AeroRP consists of two phases [2]. In the first phase, each node learns and makes a list of available neighbors at any given point in time. It utilises a number of different mechanisms to facilitate neighbor discovery, including periodic beacons and storing trajectory data in actual data packets. The second phase of the algorithm is to find the appropriate next hop to destination to forward the data packets. The location of the destination is known by all nodes, and a neighbor table is maintained that is updated based on the mechanism used in the first phase. The node uses this information to choose the next hop for each data packet. This is done by choosing the neighbor that has the smallest *time to intercept* (TTI) that indicates the time it will take for a node to be within transmission range of the destination if it continues on its current trajectory. Assume that node n_0 wants to send a data packet to the ground station D and the transmission range of all nodes is R . The TTI for each node is calculated as:

$$\text{TTI} = \frac{\Delta d - R}{s_d} \quad (1)$$

in which Δd gives the euclidean distance between the current location of a potential node and the destination node D and s_d is the relative velocity a potential neighbor has with respect to the destination. A high and positive s_d infers the neighbor is moving towards the destination at a high speed; high and negative s_d infers the neighbor is moving away.

A. Speed Component

Here, we work out exactly how s_d is calculated. Given a neighbor n_i that has geographical coordinates of x_i, y_i and a velocity of v_{xi}, v_{yi} , the velocity vector for n_i is calculated as:

$$v_i = \sqrt{v_{xi}^2 + v_{yi}^2} \quad (2)$$

The angle in degrees¹ between the positive x-axis of n_i plane and n_i velocity vector is:

$$\Theta = \text{atan2}(v_{yi}, v_{xi}) \times \frac{180}{\pi} \quad (3)$$

Destination D has geographical coordinates x_d, y_d . The angle between the positive x -axis of n_i plane and the imaginary line drawn between n_i and D is:

$$\bar{\Theta} = \text{atan2}(y_d - y_i, x_d - x_i) \times \frac{180}{\pi} \quad (4)$$

The difference between the angles ($\Theta - \bar{\Theta}$) gives the angle between n_i velocity and the imaginary line drawn between n_i and D. This gives us s_d :

$$s_d = v_i \times \cos(\Theta - \bar{\Theta}) \quad (5)$$

¹ $\text{atan2}(x, y)$ is a two-argument convenience function available in most programming languages that computes the angle in radians between the positive x-axis of a plane and the x, y coordinates provided in the arguments.

B. Refining the Time to Intercept

The *time to intercept* (TTI) is the primary metric used for routing decisions in AeroRP. A source node calculates the TTI of its neighbors to understand when its neighbors will potentially be within transmission range of the destination and make the decision to route to the neighbor that will potentially be within transmission range of the destination the soonest and thus has the lowest TTI. Given a potential neighbor n_i with coordinates x_i, y_i, z_i and a destination D with coordinates x_d, y_d, z_d , the Euclidean distance between the two is given as:

$$\Delta d = \sqrt{(x_d - x_i)^2 + (y_d - y_i)^2 + (z_d - z_i)^2} \quad (6)$$

The TTI is calculated as follows in which R is the transmission range of the mobile devices:

$$\text{TTI} = \begin{cases} 0 & \text{for } s_d < 0 \text{ and } \Delta d > R \\ \frac{\Delta d - R}{s_d} & \text{otherwise} \end{cases} \quad (7)$$

TTI=0 is a special case that indicates to never choose this neighbor as a next hop because we do not choose nodes that are moving away from the destination and not within transmission range. We use a negative TTI because this is an indication of a node being within transmission range of the destination that should be chosen as a next hop (lowest TTI). We always choose within transmission range of the destination over nodes that are not. We also choose nodes that are within transmission range of the destination but moving away from the destination to be chosen if there are no nodes within transmission range of the destination that are moving towards the destination. The nodes within transmission range moving towards the destination will be favored over those nodes within transmission range but moving away from the destination because these nodes will have a positive TTI due to a negative s_d and a negative $\Delta d - R$, and the nodes moving towards the destination will have a negative TTI.

C. Predicting Neighbors Out of Range

In a highly dynamic mobile environment in which links are constantly being broken due to high speeds, it may not be enough to just purge entries that have not been heard from based on a configurable hold time. Hence, we try to predict nodes that will be out of transmission range and remove them from next hop consideration. The predicted distance \hat{d} between n_0 and n_1 is:

$$\begin{aligned} \bar{x}_i &= x_i + v_{xi}(t_1 - t_0) \\ \bar{y}_i &= y_i + v_{yi}(t_1 - t_0) \\ \bar{z}_i &= z_i + v_{zi}(t_1 - t_0) \\ \hat{d} &= \sqrt{(x_0 - \bar{x}_i)^2 + (y_0 - \bar{y}_i)^2 + (z_0 - \bar{z}_i)^2} \end{aligned} \quad (8)$$

Finally, the logic used to predict whether or not n_i is going to be out of n_0 's range is given as:

$$\text{OutOfRange} = \begin{cases} \text{true} & \text{for } \hat{d} \geq R \\ \text{false} & \text{for } \hat{d} < R \end{cases} \quad (9)$$

IV. AeroRP FLOW

The AeroRP routing protocol has both a neighbour discovery and a data forwarding phase as previously discussed. In order to discover neighbors in *beacon* mode, nodes receive AeroRP hello beacons from their

neighbors. The node either creates a new entry in its neighbor table or updates its current data regarding the node from which it received the hello beacon. This neighbor table is used to calculate the TTI of its neighbors in order to make routing decisions.

Given the wireless nature of node communication in MANETs, it is possible for a node to be promiscuous and overhear all packets, even those packets that are not intended for a given node. In *beaconless promiscuous* mode, AeroRP takes advantage of this behavior and adds location information to each data packet per-hop as opposed to sending periodic hello beacons with this information. All nodes within transmission range, including those nodes that are not the intended receiver, can listen to the data packet and extract the location information from the header and store this location information for making routing decisions.

For the case when the node receives a packet for which the node itself has the best TTI but is not within transmission range of the destination, the packet can be queued in a configurable sized queue for a configurable amount of time. The queue is checked at a configurable frequency to see if there is a neighbor with a lower TTI than the local node. When a neighbor with a lower TTI is encountered, the packets from the queue are sent at a configurable data rate. There are currently three different AeroRP modes for when the local node has the best TTI: 1) **Ferry**: queue the packets indefinitely until a node with a lower TTI is found, 2) **Buffer**: queue the packets in a finite sized queue with a finite timeout until a node with a lower TTI is found, and 3) **Drop**: drop the packet. The flow of receiving and routing a data packet in AeroRP is illustrated in Figure 1.

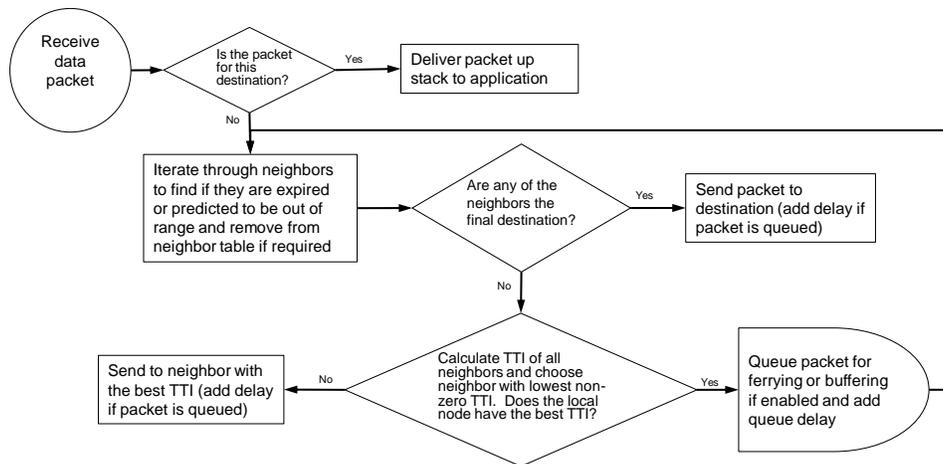


Figure 1: AeroRP flow

When receiving a data packet, a node uses its neighbor table to decide how to route the packet. If the node is not the packet’s destination, the node will clean its neighbor table of stale entries and those nodes that are predicted to be out of range as discussed in Section III.C. If one of the neighbors is the destination of the packet, the packet will be transmitted to the destination. Otherwise, the packet will be transmitted to a neighbor that has a better TTI. If the local node has the best TTI, it will ferry, buffer, or drop the packet depending on the mode that AeroRP is in.

V. SIMULATION RESULTS

In this section, we present the results of simulations conducted with the ns-3 simulator [26] to compare the performance of AeroRP and its various modes with the traditional MANET routing protocols AODV (ad-hoc on-demand distance vector) [22], DSDV (destination-sequenced distance vector) [27], and OLSR (optimized link state routing) [28, 29]. The topology setup consists of between 10 and 100 wireless TAs that are randomly distributed over a 150 km² test range. A single stationary sink node is located in the center of the simulation area representing the GS. The ANs follow a modified random-waypoint mobility model for a total of 1000 seconds with pause times of zero. Various low and high velocities are tested from 1 m/s to 1200 m/s. Each node has a transmission power configured such that it has a 15 nmi (27.8 km) transmission range. In order to evaluate the performance of the network, we send constant bit rate (CBR) traffic from all ANs to the GS at 1 pkt/s with a packet size of 1000 B. A warmup time of 100 s is used to allow the network to stabilize and the mobility model to converge. AeroRP is tested with both a ferrying mode as well as a buffer mode. In buffer mode, the packet queue size and timeout is configured to be in line with what AODV and DSDV implement. If the packet is neither ferried nor buffered, the packet is immediately dropped if there is no route. AeroRP is tested with these three modes as well as in both beacon and beaconless promiscuous mode.

In order to measure the performance of the various routing algorithms in this study, we used two metrics: *packet delivery ratio* (PDR) and *accuracy*. PDR is the number of packets received divided by the number of packets sent at the application layer. Note that not necessarily all packets sent at the application layer will be sent at the MAC layer, this can happen if there is no route for the packet. Accuracy is the number of packets received divided by the number of packets sent at the MAC layer. This allows us to measure how *accurate* a route is for a given routing protocol based on whether or not the route that was chosen for the packet results in a successful reception at the destination. This is a good metric to gauge the *quality of a route* in a highly dynamic topology, in which the validity of a route can rapidly change.

Figure 2 shows the performance in terms of PDR as the velocity exponentially increases from 1 m/s to 1000 m/s for 60 nodes. DSDV performs better than AODV but still poorly at this node density with its PDR approaching 0 as the velocity reaches 1000 m/s. OLSR performs well and in fact has higher PDR for all velocities when AeroRP is not ferrying and in beaconless promiscuous mode, and for lower velocities when AeroRP is buffering packets in beaconless promiscuous mode. However, all other AeroRP modes outperform OLSR. Most of the routing protocols' PDR performance degrades as the velocity increases with the exception of OLSR and AeroRP when buffering packets in beaconless promiscuous mode. The plot clearly shows that ferrying packets in beaconless promiscuous mode outperforms the other protocols and modes, and stays at a constant PDR when advancing from 100 m/s to 1000 m/s. The combination of some kind of packet buffering, whether it is indefinite or finite, coupled with beaconless promiscuous mode yields the best PDR.

Figure 3 shows how accuracy is effected as velocity increases exponentially from 1 m/s to 1000 m/s. OLSR, DSDV, and AODV yield an accuracy of less than 50% except for OLSR at 1000 m/s. All of the various modes of AeroRP have an accuracy of 50% or higher at all velocities. This illustrates AeroRP's ability to accurately predict the delivery of a packet based on the known transmission range and the known and predicted distance between the source and next hop. Of the AeroRP modes, the beaconless promiscuous mode is more accurate than the beacon modes. The beaconless promiscuous mode is more accurate for

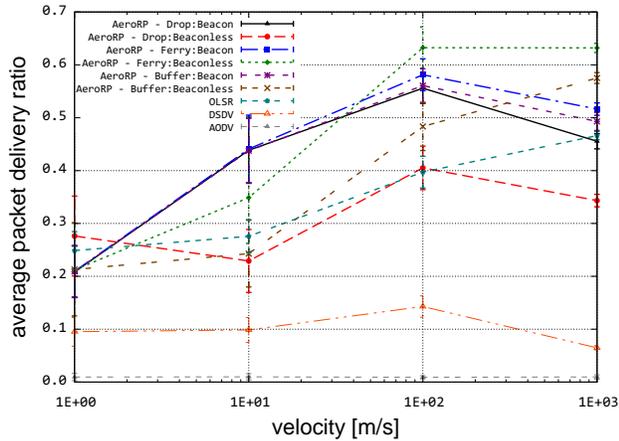


Figure 2: Effect of velocity on PDR

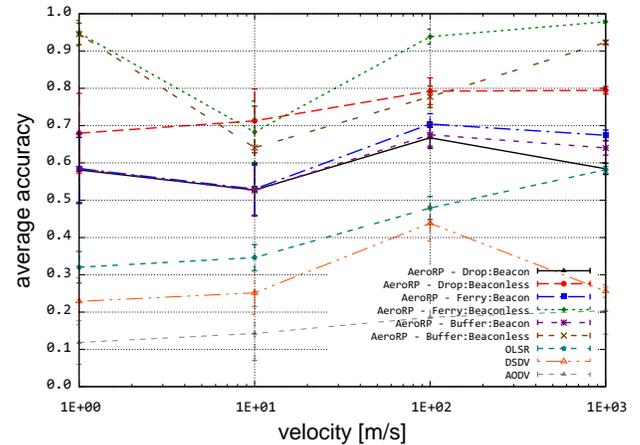


Figure 3: Effect of velocity on accuracy

two reasons. First, the surrounding nodes overhear data packets and thus trajectory data every single time a packet is transmitted. This results in sharing trajectory information more often than sending out periodic hello beacons. Second, putting the control data in the actual data packets makes the communication more symmetric than sending separate control packets. A control packet that is 44 B may be transmitted successfully to a neighbor. However, this does not necessarily mean that a 1000 B payload plus the control overhead can necessarily be successfully transmitted to that same neighbor, especially if that neighbor is on the edge of the transmission range.

Figure 4 shows the average PDR as the number of nodes are increased when traveling at 1200 m/s. The node density of the network affects all of the routing protocols with AeroRP ferrying packets in beaconless promiscuous mode performing the best. The PDR for all AeroRP modes increases as the number of nodes increase with the exception of a slight performance degradation as the number of nodes approaches 90 and higher. The PDR for both DSDV and AODV immediately degrades as the number of nodes increases. This is most likely due to the increase in overhead observed as the number of nodes increases. The performance of OLSR starts to degrade around 50 nodes. This suggests, that as the number of nodes increase, AeroRP is able to make more intelligent decisions on how to move the data packets towards the destination whereas the non-AeroRP routing protocols are relying on non-geographic based links to move the packet to the destination.

Figure 5 shows how accurate the various routing protocols are as the number of nodes increase when traveling at 1200 m/s. The accuracy of ferrying and buffering packets with AeroRP stays constant at almost 100% as the number of nodes increases. This high accuracy is due to the same reasons previously discussed when the velocities are increasing. All of the AeroRP modes have an accuracy of 50% or higher with the accuracy increasing as the number of nodes increase with the exception of AeroRP running in beaconless promiscuous mode but with no ferrying or buffering of packets. This decrease in accuracy can be attributed to the nodes having to rely on data transmissions to communicate their trajectories to nearby nodes. The buffering and ferrying allows data to be delivered at different times in the simulation whereas AeroRP that is not ferrying or buffering packets is not sharing this information as often. This does not occur when AeroRP is in beacon mode but not ferrying or buffering packets because it still regularly

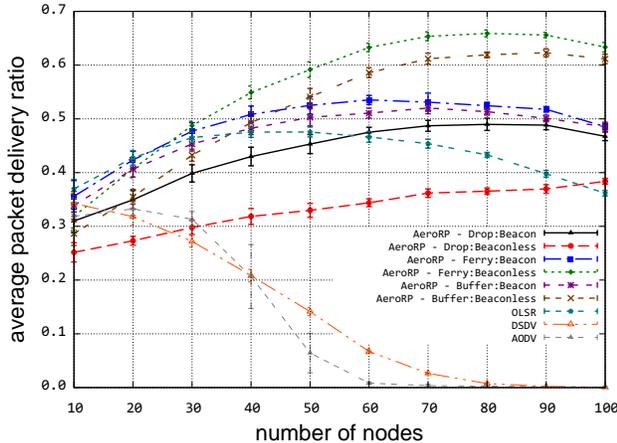


Figure 4: Effect of node density on PDR

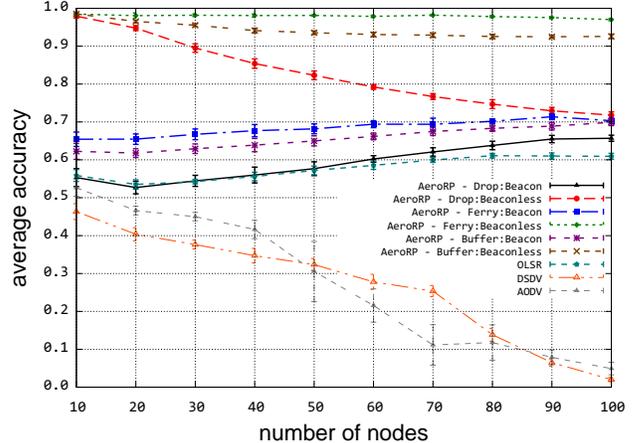


Figure 5: Effect of node density on accuracy

shares its trajectory information with its neighbors in the form of periodic hello beacons. OLSR yields higher accuracy as the number of nodes increases but still not as high as the AeroRP modes. The accuracy of DSDV and AODV actually decreases as the number of nodes increases, probably due to the increase in overhead as the number of nodes increases.

VI. CONCLUSIONS AND FUTURE WORK

Emerging networked telemetry systems require domain-specific protocols for communication to cope with the challenges faced in highly-dynamic environments. In this paper, we present *decision metrics* that AeroRP utilises to route packets to destination. Furthermore, we present the *preliminary* results of AeroRP performance against other legacy MANET routing protocols in ns-3 simulator in realistic high-velocity scenarios. Our results indicate that AeroRP generally outperforms when used in combination with packet buffering, whether indefinite or finite, coupled with beaconless promiscuous mode. Moreover, performance results also suggest that AeroRP can more *accurately* predict the delivery of a packet, based on the known transmission range, the known and predicted distance between the source and next hop, and the velocity of the next hop.

Our future work includes testing the AeroRP with a Gauss-Markov mobility model [30] with memory-based movement, which is more suitable for iNET scenarios. We are also working on simulating the AeroRP routing protocol with the AeroTP transport protocol [31] to analyse their performance when combined, including the effects of buffering modes on transport layer operation.

ACKNOWLEDGEMENTS

The authors would like to thank the Test Resource Management Center (TRMC) Test and Evaluation/Science and Technology (T&E/S&T) Program for their support. This work was funded in part by the T&E/S&T Program through the Army PEO STRI Contracting Office, contract number W900KK-09-C-0019 for AeroNP and AeroTP: Aeronautical Network and Transport Protocols for iNET (ANTP). This work was also funded in part by the International Foundation for Telemetry (IFT). We would like to acknowledge the ResiliNets group members Justin P. Rohrer for useful discussions on this work. We would

also like to thank Kip Temple and the membership of the iNET working group for discussions that led to this work.

REFERENCES

- [1] iNET Working Group, “<http://www.inetprogram.org>.”
- [2] A. Jabbar and J. P. Sterbenz, “AeroRP: A Geolocation Assisted Aeronautical Routing Protocol for Highly Dynamic Telemetry Environments,” in *Proceedings of the ITC*, (Las Vegas, NV), October 26–29 2009.
- [3] “iNET Needs Discernment Report, version 1.0.” CTEIP, May 2004.
- [4] “iNET Technology Shortfalls Report, version 1.0.” CTEIP, July 2004.
- [5] “iNET System Architecture, version 2007.1.” CTEIP, July 2007.
- [6] M. Mauve, A. Widmer, and H. Hartenstein, “A Survey on Position-Based Routing in Mobile Ad Hoc Networks,” *IEEE Network*, vol. 15, no. 6, pp. 30–39, 2001.
- [7] C. Lemmon, S. M. Lui, and I. Lee, “Geographic Forwarding and Routing for Ad-hoc Wireless Network: A Survey,” *NCM*, vol. 0, pp. 188–195, 2009.
- [8] M. de la Fuente and H. Ladiod, “A Performance Comparison of Position-Based Routing Approaches for Mobile Ad Hoc Networks,” *IEEE VTC*, pp. 1–5, October 2007.
- [9] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, “A Distance Routing Effect Algorithm for Mobility (DREAM),” in *MobiCom '98*, (New York, NY, USA), pp. 76–84, ACM, 1998.
- [10] Y.-B. Ko and N. H. Vaidya, “Location-aided Routing (LAR) in Mobile Ad Hoc Networks,” in *MobiCom '98*, (New York, NY, USA), pp. 66–75, ACM, 1998.
- [11] B. Karp and H. T. Kung, “GPSR: Greedy Perimeter Stateless Routing for Wireless Networks,” in *MobiCom '00*, (New York, NY, USA), pp. 243–254, ACM, 2000.
- [12] A. Capone, L. Pizziniaco, I. Filippini, and M. de la Fuente, “A SiFT: an Efficient Method for Trajectory Based Forwarding,” in *2nd International Symposium on Wireless Communication Systems*, pp. 135 – 139, Sept. 2005.
- [13] R. L. Lidowski, B. E. Mullins, and R. O. Baldwin, “A Novel Communications Protocol using Geographic Routing for Swarming UAVs Performing a Search Mission,” in *Proc. of IEEE PERCOM*, pp. 1–7, 2009.
- [14] E. Kuiper and S. Nadjm-Tehrani, “Geographical Routing in Intermittently Connected Ad Hoc Networks,” in *IEEE AINI*, pp. 1690–1695, March 2008.
- [15] B. Zhou, Y.-Z. Lee, and M. Gerla, “Direction Assisted Geographic Routing for Mobile Ad Hoc Networks,” in *Military Communications Conference*, pp. 1–7, Nov. 2008.
- [16] Z. Jin, N. Yan, and L. Bing, “Reliable On-Demand Geographic Routing Protocol Resolving Network Disconnection for VANET,” in *WiCOM*, pp. 1–4, Sept. 2009.
- [17] S. Jung, D. Lee, S. Yoon, J. Shin, Y. Lee, and J. Mo, “A Geographic Routing Protocol Utilizing Link Lifetime and Power Control for Mobile Ad Hoc Networks,” in *Proc. of ACM FOWANC*, pp. 25–32, 2008.
- [18] Q. J. Chen, S. Kanhere, M. Hassan, and K.-C. Lan, “Adaptive Position Update in Geographic Routing,” in *IEEE International Conference on Communications*, vol. 9, pp. 4046–4051, June 2006.

- [19] J. Na and C. kwon Kim, “GLR: A novel geographic routing scheme for large wireless ad hoc networks,” *Computer Networks*, vol. 50, no. 17, pp. 3434 – 3448, 2006.
- [20] J. Li and S. Shatz, “Toward Using Node Mobility to Enhance Greedy-forwarding in Geographic Routing for Mobile Ad Hoc Networks,” in *MODUS 2008*, pp. 1–8, April 2008.
- [21] M. Iordanakis, D. Yannis, K. Karras, G. Bogdos, G. Dilintas, M. Amirfeiz, G. Colangelo, and S. Baiotti, “Ad-hoc Routing Protocol for Aeronautical Mobile Ad-Hoc Networks,” in *CSNDSP*, pp. 543–547, 2006.
- [22] C. Perkins, E. Belding-Royer, and S. Das, “Ad Hoc On-Demand Distance Vector (AODV) Routing.” RFC 3561 (Experimental), July 2003.
- [23] N. K. E. Sakhaee, A. Jamalipour, “Aeronautical Ad Hoc Networks,” in *IEEE Wireless Communications and Networking Conference*, vol. 1, pp. 246 –251, April 2006.
- [24] F. Tchakountio and R. Ramanathan, “Anticipatory Routing for Highly Mobile Endpoints,” in *WMCSA '04*, (Washington, DC, USA), pp. 94–101, IEEE Computer Society, 2004.
- [25] F. Tchakountio and R. Ramanathan, “Tracking Highly Mobile Endpoints,” in *WOWMOM '01*, (New York, NY, USA), pp. 83–94, ACM, 2001.
- [26] “The ns-3 Network Simulator.” <http://www.nsnam.org>, July 2009.
- [27] C. E. Perkins and P. Bhagwat, “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers,” in *SIGCOMM '94*, (New York, NY, USA), pp. 234–244, ACM, 1994.
- [28] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, “Optimized Link State Routing Protocol for Ad Hoc Networks,” in *IEEE INMIC 2001*, pp. 62–68, August 2002.
- [29] T. Clausen and P. Jacquet, “Optimized Link State Routing Protocol (OLSR).” RFC 3626 (Experimental), Oct. 2003.
- [30] D. Broyles, A. Jabbar, and J. P. G. Sterbenz, “Design and analysis of a 3–D gauss-markov mobility model for highly-dynamic airborne networks,” in *Proc. of the ITC*, (San Diego, CA), October 2010. (to appear).
- [31] K. S. Pathapati, J. P. Rohrer, and J. P. G. Sterbenz, “Edge-to-edge arq: Transport-layer reliability for airborne telemetry networks,” in *Proc. of the ITC*, (San Diego, CA), October 2010. (to appear).