AeroRP: A Geolocation Assisted Aeronautical Routing Protocol for Highly Dynamic Telemetry Environments

Abdul Jabbar and James P.G. Sterbenz
Department of Electrical Engineering & Computer Science
The University of Kansas
Lawrence, KS 66045
{jabbar,jpgs}@ittc.ku.edu

ABSTRACT

With the increasing importance of networked systems for telemetry, there is a need for efficient routing algorithms in aeronautical environments. Unlike traditional mobile networks, the highly dynamic nature of airborne networks results in extremely short-lived paths, especially for multi-hop scenarios thereby necessitating domain-specific protocols. In this paper, we present the detailed design and evaluation of AeroRP, a cross-layered routing protocol designed specifically for airborne telemetry applications. AeroRP exploits the broadcast nature of the wireless medium along with the physical node location and trajectory to improve the data delivery in Mach-speed mobile scenarios. We present a multi-modal protocol that addresses various operational scenarios of test and telemetry networks. Preliminary simulation results show that AeroRP significantly outperforms traditional MANET routing protocols while limiting the overhead.

INTRODUCTION

Airborne Telemetry has evolved significantly from point-to-point communication to highly networked systems. However, certain challenges still remain in the extreme case of airborne test and evaluation scenarios in which the node speeds are in excess of Mach 3. Besides the node mobility, other limitations such as short transmission ranges and varying channel characteristics often result in unpredictable and intermittent connectivity. Furthermore, a typical telemetry network (e.g. [1]) consists of nodes that differ in several key communication characteristics resulting in heterogeneous environments.In order to illustrate these challenges, we use the example of the telemetry networks designed for *Major Range and Test Facility Bases* [2]. The key test elements, called *test articles* (TAs) are test aircrafts operating at speeds up to Mach 3.5. They have limited transmission ranges and are the source of the telemetry data which is destined to *ground stations* (GS) with directional antennas. The *relay nodes* are special purpose nodes deployed to improve the network connectivity. The intermittent connectivity in this environment is characterized by contact durations that are in the order of few seconds. It has been argued [1] that such networks benefit from a domain specific architecture that is designed to facilitate cross-layering with efficient geolocation assisted routing.

Current routing protocols are not designed to address the needs of telemetry applications [3] and there remain a number of issues to be solved [4]. Reactive routing protocols such as AODV [5] and DSR [6] attempt to construct source-to-destination paths on demand and are not suitable because of the delay involved in finding paths and because such paths may not be valid for very long in a highly-dynamic network. On the other hand, proactive routing protocols such as DSDV [7] and OLSR [8] forward packets on a hop-by-hop basis and depend on global route convergence. This generates excessive overhead due to frequent route updates (assuming convergence is even possible) and is not suitable for a bandwidth-constrained telemetry network. Some of the existing protocols that forward packets one hop at a time without attempting to construct the entire path are based on node location and movement [9, 10, 11, 12, 13]. However, there are not suitable for extremely dynamic connectivity resulting from node speeds as high as Mach 7. Furthermore, existing protocols are optimized for a single mode of operational characteristics, which is not suitable for test and telemetry networks in which the operational conditions and service requirements vary widely from one test to anther.

A. ANTP Approach

While telemetry networks pose challenges to traditional wired and wireless routing protocols, they also facilitate the use of cross-layer information such as the knowledge of the airborne node location and trajectory by domain specific protocols. This paper presents the design and evaluation of *AeroRP* – an efficient routing protocol introduced in [1] and [14], that exploits location information available in the test environment with the use of a domain specific network protocol AeroNP [1] to mitigate the short contact times of high-velocity nodes. However, explicit information of node location cannot always be assured in testing of aircraft, hence the need for the protocol to be modular and opportunistic in terms of adapting to the limitations of individual test cases.

The rest of the paper is organized as follows: section presents specific challenges to routing in the aeronautical environment through a case study of the iNET scenario. In section A. we present AeroRP: a lightweight routing protocol leveraging node location information. Finally in section B., we present simulation results comparing AeroRP to traditional MANET protocols in a highly dynamic network.

AeroRP: LOCATION-AWARE HIGHLY ADAPTIVE ROUTING ALGORITHM

The small contact duration among TAs results in frequent routing changes and is indicative of the need for an intelligent multihop routing protocol, supporting reliable communication over the highly dynamic physical topology. As discussed previously, existing routing mechanisms generate significant overhead and do not converge quickly (if ever) in the presence of frequent topology changes and hence are not suitable for telemetry networks. The AeroRP routing protocol is specifically designed to address the issues related to highly mobile aeronautical environments. We utilise a number of mechanisms that have been researched independently for use in environments with characteristics similar to those of aeronautical telemetry:

1. **Proactive behavior:** AeroRP is a fundamentally proactive routing protocol, but with limited updates, and therefore low protocol overheard.
2. **Exploits cross-layer data:** AeroRP is designed to exploit the explicit cross-layering support provided by AeroNP and the geographic node location and trajectory information available at nodes.
3. **Per-hop behavior:** Unlike existing protocols, AeroRP forwards data per-hop based on partial local information and routes thereby avoiding the necessity for global convergence, making it especially suitable for highly-dynamic environments.
4. **Multi-modal:** Telemetry applications present a high level of variation in their operational parameters. For example, based on the security requirements of the test application, the geolocation of the nodes may or may not be available. In order to support these dynamics in operation, policies, and constraints, AeroRP provides multiple modes of operation.

AeroRP is based on the ANTP [1] system architecture that uses a cross-layered network protocol designed for this environment: AeroNP. The AeroNP protocol header includes fields for source and destination locations. It also has two fields to specify the quality of service of packets in the network: data *type* (e.g. command and control, telemetry) and *priority* with in a given type.

B. Protocol Operation

The basic operation of AeroRP consists of two phases. 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, discussed later in this section. The second phase of the algorithm is to find the appropriate next hop to forward the data packets. In order to forward the packets toward a specific destination, additional information such as location data or route updates is required. For each of these two phases the protocol defines a number of different mechanisms. The particular choice of mechanism to be used is dependent upon the mode of operation. The protocol does not specify a predefined set of discrete operational modes; the total number of supported modes is merely the combination of all the different mechanisms available. We now consider each of the two phases in more detail:

1. Neighbor Discovery**:** The first objective of a node in the telemetry network is to determine its neighboring nodes. In order to achieve this, we use several different mechanisms with the objective to minimise overhead and increase adaptability. One or more of the following mechanisms may be used to populate the forwarding table depending upon the operational constraints.

- ***Active snooping*** is the primary mechanism used by the node to locate and identify its neighbors. In a wireless TDMA network, a node that is not transmitting listens to all transmissions on the wireless channel. AeroRP adds the transmitting MAC address of each overheard packet to its neighbor table. The protocol assumes cooperative nodes and symmetric transmission ranges. This implies that if a node can hear transmissions from a node, it can also communicate with that node. Stale entries are removed from the neighbor table if no transmissions from a node are heard for a predetermined time interval related to the anticipated contact duration.

- ***Hello beacons*** are used by idle nodes to advertise their presence. When neighboring nodes hear a hello beacon, they update their neighbor table appropriately. The frequency of the hello beacon is related to the minimum contact duration. For example, if the contact duration is 10 sec, the hello beacon is transmitted every second.

- ***Ground station updates*** may used to augment or replace *active snooping* in some of the telemetry scenarios, in which the ground station has a partial or even complete mission plan. The ground station sends periodic updates containing the location and trajectory vectors predicted by the mission plan to all nodes.

Security requirements may impose certain restrictions in the Aeronautical networks. In certain cases where node location or trajectory is considered sensitive, individual nodes may not include this information in the header of data packets or hello updates. In this case, the ground station may send location updates of all nodes on an encrypted channel. Finally, in the most secure mode, no geographic node information is available and the routes have to built using traditional MANET methods such as explicit routing updates and exchange of node contacts between the neighbors.

Given the dynamic nature of the aeronautical network, neighbor discovery not only consists of finding nodes within transmission range, but also determining the duration for which a discovered node will remain within range. Depending upon operational constraints, this information is obtained via different mechanisms: location and trajectory information is included in the network protocol (AeroNP) header [1], or in updates sent by the ground station.

2. Data Forwarding**:** After neighbor discovery, the second phase of AeroRP is for individual nodes to determine the next hop for a particular transmission. Recall that, unlike conventional protocols, AeroRP performs hop-by-hop forwarding based on partial paths without the full knowledge of the end-to-end paths. Each node forwards packets such that they end up geographically closer to the destination, which will frequently be a GS in the telemetry environment.

When any given node needs to transmit telemetry data, and assuming that one or more neighbors are discovered, the data packets are forwarded to the node that is nearest to the destination as calculated from its current coordinates and trajectory. The destination location is obtained in a manner similar to that of discovering neighbors. Furthermore, in a majority of test cases, the destination is the stationary ground station whose coordinates are known to all TAs. The actual algorithm for finding the best node to forward (or handover) the data packet is given in Section B.

In order to avoid congestion at any given node, AeroRP utilizes the *congestion indicator* [15, 16] field of the AeroNP header. Each node uses the CI field to indicate its own congestion level. All packet transmissions from a node carry the CI field along with the type and priority of the data. All the neighboring nodes are thus made aware of the congestion at a given node for a given priority of the traffic and refrain from forwarding equal or lower priority traffic to the congested node.

AERORP ALGORITHM

Let the position of i^{th} airborne node, n_i be represented by the vector $\mathbf{P}_i = (x_i, y_i, z_i)$ and the trajectory is defined by the vector $\mathbf{T}_i = (s_i, \theta_i, \phi_i)$, where x, y, and z are the absolute node coordinates, $\mathbf{T}$ is the spherical direction vector (seed, inclination, and azimuth). Since the network is highly dynamic both the position and trajectory of nodes is time dependent. For a given source–destination pair, at a given time t, let the source node n_s has the position be $\mathbf{P}_s^t = (x_s^t, y_s^t, z_s^t)$ and the trajectory be $\mathbf{T}_s^t = (s_s^t, \theta_s^t, \phi_s^t)$. Similarly the destination node n_d is represented by $\mathbf{P}_d^t = (x_d^t, y_d^t, z_d^t)$ and $\mathbf{T}_d^t = (s_d^t, \theta_d^t, \phi_d^t)$. If the destination happens to be a stationary ground station, then $\mathbf{P}_d^t = \mathbf{P}_d, \forall t$. Finally, let the congestion status of the node be given by the vector $\mathbf{C}_i = \{CI, \text{priority}\}$, where *CI* and *priority* are the congestion indicator and priority fields that are extracted from the AeroNP header.

Step 1: Each node maintains two tables: a *neighbor table* that stores the information about the nodes that are currently in the transmission range, and *destination data table* that stores the information all destinations, which may or may not be in the currently in the transmission range. Initially, let the number of neighbors represented by the neighbor list N be zero, i.e., $N = \emptyset$.

Step 2: When the node receives any transmission, it updates the neighbor and destination data tables. If the captured packet is an overheard transmission or hello advertisement from node n_i, the node i is assumed to be in the transmission range of the current node. Hence the neighbor list is updated as $N = N \cup \{n_i\}$. Furthermore, the macID, position, trajectory, and congestion status of the node are derived from its header and stored in the neighbor table as the tuple $\{\text{macID}_i, \mathbf{P}_i^t, \mathbf{T}_i^t, \mathbf{C}_i^t\}$. If the received transmission is a *ground station update*, each entry in the update is stored in the destination data table as the tuple $\{\text{time}, \text{macID}_i, \mathbf{P}_i^t, \mathbf{T}_i^t, \mathbf{C}_i^t\}$. Since the GS update may contain information on node positions in future, the entries in the destination data table are time stamped. Lastly, when a ground station update is received, the location and trajectory fields of neighbor table entries are updated with the latest values.

Step 3: At the completion of step 2, assume that a given node n_0 has k discovered neighbors. Hence $N_0 = \{n_0, n_1, \ldots, n_i, \ldots, n_{k-1}\}$. Note that each node treats itself as the first neighbor. Assume that this node n_0 wants to send a data packet to the ground station n_d with position $\mathbf{P}_d$. Assume that the transmission range of all nodes is R. Next, we calculate the *time to intercept*, TTI for all neighbors. The TTI_i represents the time it will take for node n_i to get reach within the transmission range of the destination if it continues on its current trajectory. TTI is calculated as:

$$\text{TTI}_i = \frac{|\mathbf{P}_d^t - \mathbf{P}_i^t| - R}{s_d} \tag{1}$$

where $|\mathbf{P}_d^t - \mathbf{P}_i^t|$ gives the euclidian distance between the current location of node n_i and the destination node n_d and s_d is the component of the actual speed r of node n_i in the direction of the destination.

Step 4: Finally, the data is forwarded to the j^{th} node, n_j such that:

$$\text{TTI}_j = \min\{\text{TTI}_i\} \;\; \forall i : n_i \in N_0 \tag{2}$$

The process is repeated at every node, until the data reaches the destination.

SIMULATION RESULTS

In this section, we presents the results of simulations conducted in ns-2 simulator [17] to compare the performance of the AeroRP with traditional MANET routing protocols AODV (Ad-hoc On-demand Distance Vector) and DSDV (Destination-Sequenced Distance Vector). The topology setup consists of 60 wireless TA (test article) nodes that are randomly distributed over a 150 km $\times$ 150 km test range, and a single stationary sink node is located in the center of the simulation area representing the GS. The 60 TAs follow a modified random-waypoint movement model for a total of 2000 seconds with pause times of zero. Two different test cases are simulated: In the first case each node's speed is randomly selected to be between Mach 0.3 and Mach 3.5 (100 to 1200 m/s) for each leg of the random-waypoint movement; in the second case the nodes always move at Mach 3.5. Each node has an omnidirectional antenna with a maximum range of 15 nautical miles (27.8 km).

In order to evaluate the performance of the network, we send send constant bit rate (CBR) traffic from all TA to the GS at a constant 0.2 Mb/s and a packet size of 1000-byte packets. The wireless channel rate is 11 Mb/s that far exceeds the overall load on the network. A warmup time of approximately 1000 seconds is used to allow the network to stabilise.

C. *Performance Results*

The packet delivery rate (for an aggregate source rate of 1500 packets per second) for these three protocols when the speed is varied between Mach 0.3 and Mach 3.5 is shown in Figure 1. In this first

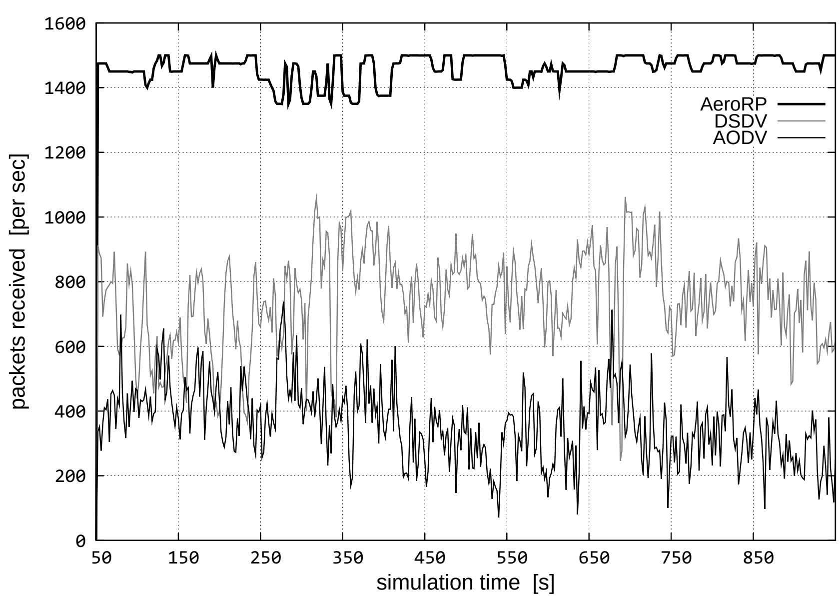

Figure 1: Packet delivery rate for Mach 0.3 to 3.5

case using AODV, only 3.25×10^5 out of 1.35×10^6 packets (24%) are received by the sink node. DSDV

performs better with 6.74×10^5 packets (50%) being received. With AeroRP, 1.31×10^6 packets (97%) are received at the sink node. Figure 2 shows the results for the second case in which nodes move at a constant speed of Mach 3.5. In this case AODV received 3.17×10^5 packets (23%), DSDV received 5.54×10^5 packets (41%), and AeroRP received 1.32×10^6 packets (97%).

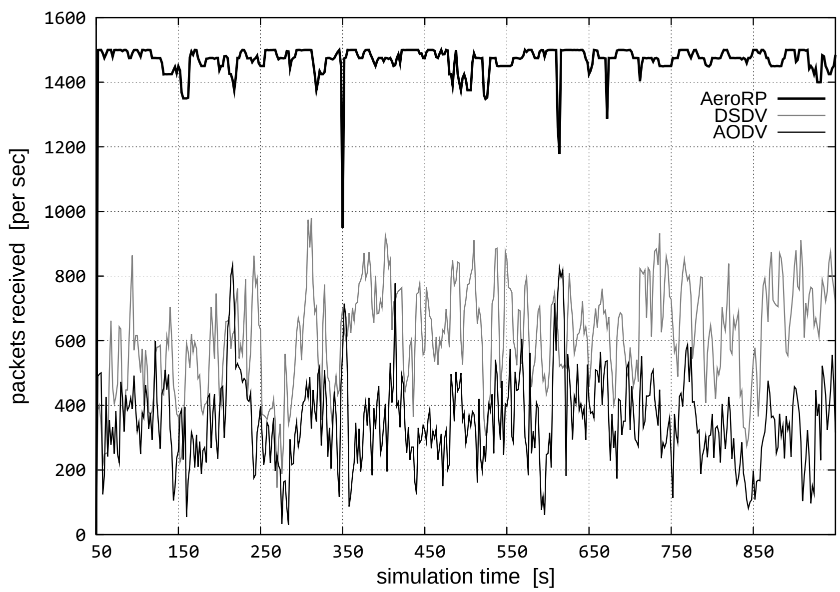

Figure 2: Packet delivery rate for Mach 3.5

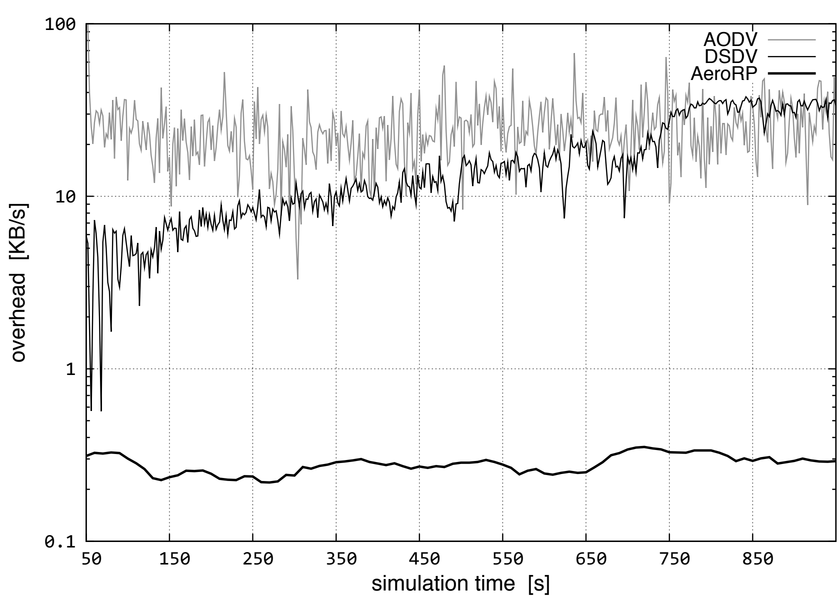

Figure 3: Routing overhead for Mach 0.3 to 3.5

The overhead incurred by AODV and DSDV for both cases is plotted in Figures 3 and 4 in terms of aggregate bytes transmitted per second. AODV incurs greater overhead due to the fact that it is on-demand

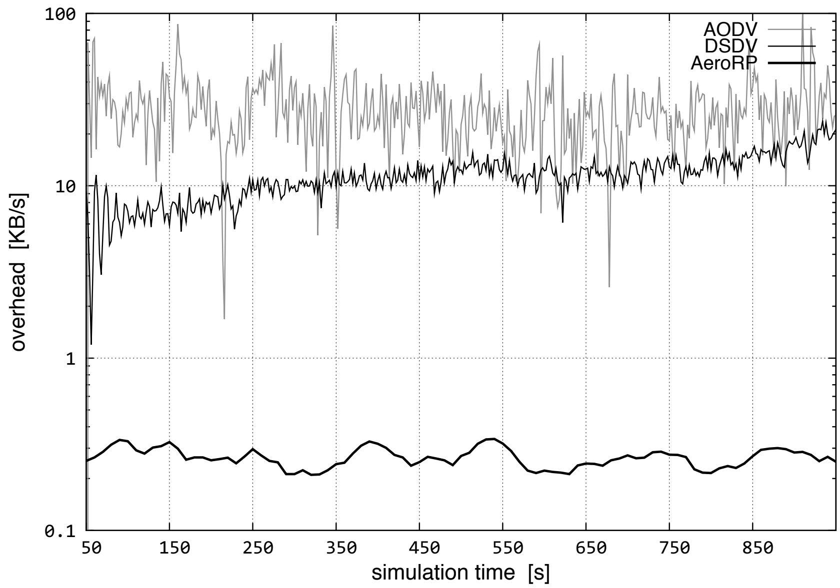

Figure 4: Routing overhead for Mach 3.5

and therefore its overhead is directly proportional to the mobility. DSDV incurs less overhead than AODV because of the periodic nature of its update messages, which are not mobility dependent. However, as the simulations progress DSDV is unable to converge due to the highly dynamic topology and generates increasing numbers of update messages. The overhead incurred by AeroRP varies depending on the type of route updates used to populate the routing table. These simulations focus on the snooping (with hello beacons) and GS broadcast mechanisms described in section A.. Snooping alone does not cause *any* overhead, because the traffic model is heavy enough that no hello messages are required. Figures 3 and 4 show the overhead induced by using the ground station to broadcast the current and predicted link-state table to all the nodes at 10-second intervals. In either case the evaluations show that the AeroRP overhead is much lower than AODV or DSDV since it does not transmit event-based updates.

Based on our examinations of the simulation trace data, the poor performance of AODV and DSDV is caused by the timescale on which they operate. In both cases they can take 30 seconds to 5 minutes to determine that a route has failed and reroute [18]. In an environment in which paths may only be stable for a few seconds, these protocols simply cannot keep up. While it is possible to minimise their route convergence time using modification such as shorter update intervals and faster dead-link detection, this would inevitably lead to increased overhead.

CONCLUSIONS AND FUTURE WORK

In this paper, we evaluate a new routing protocol AeroRP that uses geographic node location and trajectory to improve the performance of highly dynamic telemetry networks. By predicting when links will be available based on trajectory information, as well as actively listening for nearby nodes, AeroRP can send data opportunistically towards its destination and make much more efficient use of available network capacity. Simulations results show that AeroRP performs significantly better in terms of packet delivery ration and overhead when compared to AODV and DSDV.

ACKNOWLEDGEMENTS

This work was funded in part by the International Foundation for Telemetering (IFT). We would like to acknowledge the ResiliNets group members Justin P. Rohrer and Egemen K. Çetinkaya 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] A. Jabbar, E. Perrins, and J. P. G. Sterbenz, "A Cross-Layered Protocol Architecture for Highly-Dynamic Multihop Airborne Telemetry Networks," in *Proceedings of the International Telemetering Conference*, (San Diego, CA), October 27–30 2008.

[2] "iNET System Architecuture, version 2007.1." Central Test and Evaluation Investment Program (CTEIP), July 2007.

[3] "iNET Needs Discernment Report, version 1.0." Central Test and Evaluation Investment Program (CTEIP), May 19 2004.

[4] "iNET Technology Shortfalls Report, version 1.0." Central Test and Evaluation Investment Program (CTEIP), July 2004.

[5] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing." RFC 3561 (Experimental), July 2003.

[6] D. Johnson, Y. Hu, and D. Maltz, "The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4." RFC 4728 (Experimental), Feb. 2007.

[7] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *SIGCOMM '94: Proceedings of the conference on Communications architectures, protocols and applications*, (New York, NY, USA), pp. 234–244, ACM, 1994.

[8] T. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR)." RFC 3626 (Experimental), Oct. 2003.

[9] W.-H. Liao, J.-P. Sheu, and Y.-C. Tseng, "GRID: A fully location-aware routing protocol for mobile ad hoc networks," *Telecommunication Systems*, vol. 18, no. 1-3, pp. 37–60, 2001.

[10] M. de la Fuente and H. Ladiod, "A performance comparison of position-based routing approaches for mobile ad hoc networks," *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pp. 1–5, 30 2007-Oct. 3 2007.

[11] L. Galluccio, A. Leonardi, G. Morabito, and S. Palazzo, "A MAC/routing cross-layer approach to geographic forwarding in wireless sensor networks," *Ad Hoc Netw.*, vol. 5, no. 6, pp. 872–884, 2007.

[12] M. Mauve, A. Widmer, and H. Hartenstein, "A survey on position-based routing in mobile ad hoc networks," *Network, IEEE*, vol. 15, no. 6, pp. 30–39, 2001.

[13] M. Yuksel, R. Pradhan, and S. Kalyanaraman, "An implementation framework for trajectory-based routing in ad hoc networks," *Ad Hoc Networks*, vol. 4, no. 1, pp. 125–137, 2006.

[14] J. P. Rohrer, A. Jabbar, E. Perrins, and J. P. Sterbenz, "Cross-layer architectural framework for highly-mobile multihop airborne telemetry networks," in *Proc. IEEE MILCOM2008*, (San Diego, CA, USA), November 2008.

[15] K. K. Ramakrishnan and R. Jain, "A binary feedback scheme for congestion avoidance in computer networks," *ACM Transactions on Computer Systems*, vol. 8, no. 2, pp. 158–181, 1990.

[16] M. Mamun-Or-Rashid, M. M. Alam, M. A. Razzaque, and C. S. Hong, "Congestion avoidance and fair event detection in wireless sensor network," *IEICE Trans Commun*, vol. E90-B, no. 12, pp. 3362–3372, 2007.

[17] "The network simulator: ns-2." `http://www.isi.edu/nsnam/ns/`, December 2007.

[18] J. Broch, D. Maltz, D. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," *MobiCom '98: Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, pp. 85–97, Oct 1998.