

# Air to Air Communication Protocol

Arjan Durrese<sup>1</sup>, Vamsi Paruchuri<sup>1</sup>, Leonard Barolli<sup>2</sup> and Raj Jain<sup>3</sup>

<sup>1</sup>Department of Computer Science, Louisiana State University  
298 Coates Hall, Baton Rouge, LA 70803, USA

<sup>2</sup>Department of Information and Communication Engineering, Fukuoka Institute of Technology (FIT),  
3-30-1 Wajiro-Higashi, Higashi-ku, Fukuoka 811-0295, Japan  
81-92-606-4970, barolli@fit.ac.jp

<sup>3</sup>Department of Computer Science and Engineering, Washington University in St. Louis  
One Brookings Drive, St. Louis, MO 63130, USA  
314-935-4963, jain@cse.wustl.edu

*Abstract*—We present Air to Air Communication (AAC)<sup>1,2</sup>, a wireless protocol designed for communication among airplanes as well as airplanes and control centers. AAC enables the broadcast of emergency and surveillance information such as realtime video over the network even in presence of adverse conditions such as coordinated terrorist attacks. AAC is very robust. AAC minimizes the number of retransmissions and therefore reduce the collisions, which could considerably delay transmissions and disrupt the communications during emergency situations. AAC performs very well in highly dynamic ad hoc networks of airplane. AAC has the potential to significantly enhance the security of the homeland by closely monitoring the airplane which, if hijacked by terrorists or criminals, could be used as weapons. We have evaluated AAC through analysis and ns-2 simulations.

## TABLE OF CONTENTS

1. INTRODUCTION.....	1
2. BACKGROUND.....	2
3. AIR TO AIR COMMUNICATION.....	3
4. SIMULATIONS .....	5
5. CONCLUSIONS .....	7
6. ACKNOWLEDGEMENTS .....	7
REFERENCES .....	7
BIOGRAPHY .....	8

## 1. INTRODUCTION

Communication among airplanes is becoming an important tool that can guarantee the safety of flights. The increase of airplanes density accompanied with more freedom in choosing the flight path requires airplanes to be able to communicate with each other. Furthermore the threat of combined terrorist attacks, which could target more than one airplane as well as their communication capabilities, requires the development of efficient and robust protocols able to function in adverse conditions.

While in the air, the pilots communicate with the ground controllers or other airplanes using wireless channels. For this purpose, the U.S. government has allocated certain

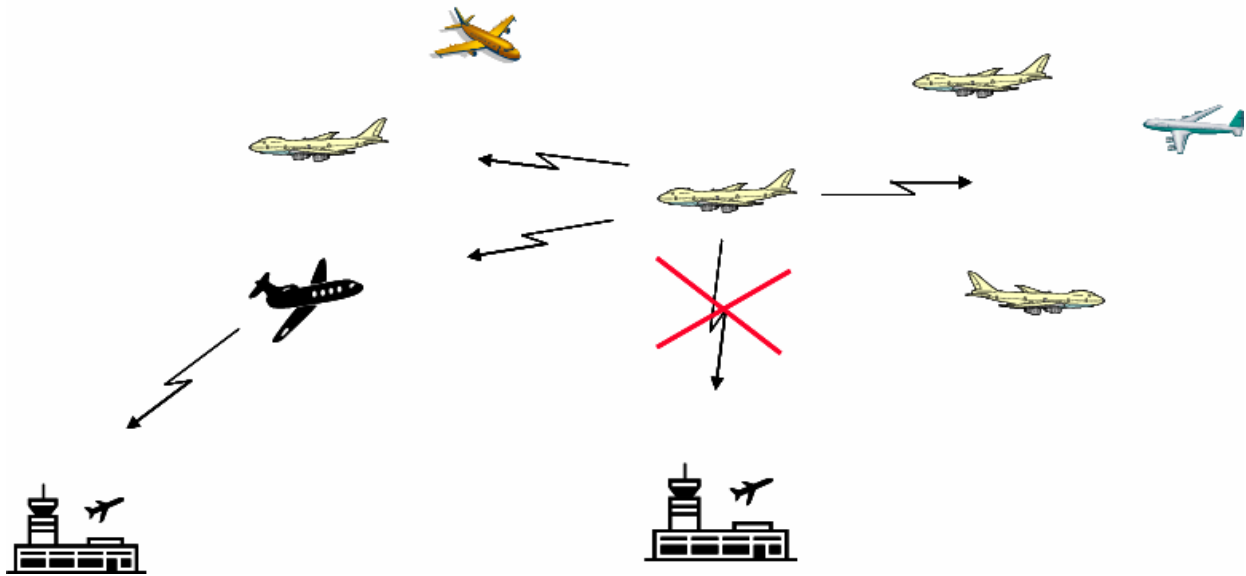
frequency and bandwidth for air-ground and air-air communications. These existing communications are highly dynamic i.e., the airplane has to search for the closest point of contact for effective communication, while today's airplanes move at supersonic speeds. The bandwidth of the available channels is very limited which allows only the radiotelephones. But the major problem of existing communications systems is that they are not very suitable for emergency situations, when the whole communication system might be under multiple attacks such as jamming of links to ground stations, disruption caused by enemy flying objects, etc.

We think that broadcast is the most suitable type of protocol for emergency situations. Ad Hoc Broadcast does not rely on any infrastructure such as specific ground station, which can become vulnerable to enemy attacks. On the other hand Broadcast is the fastest way to spread emergency information to all interested airplanes.

We present Air to Air Communication (AAC) that enables optimized broadcast among ad hoc networks of airplanes. The key feature of AAC is its simplicity that enables robustness. AAC does not require an airplane to have neighbor knowledge, thus no *Hello Messages* are needed. The only communication overhead AAC imposes are two location fields in broadcasted message and the computational overhead is minimal. We also provide extensive analytical and simulation results to study the performance of AAC. We show that AAC reduces collision among transmissions, therefore enabling emergency messages to reach their destination fast. One major advantage of AAC against other broadcast protocols is that AAC's performance is almost independent from airplanes speed. At best of our knowledge, this is the first broadcast protocol designed for air to air communications. The paper is organized as follows: Section 2 presents a review of broadcast protocols presented for ad hoc networks. Section 3 presents the AAC protocol. Extensive simulation results are presented in Section 4 and finally Section 5 concludes.

<sup>1</sup> 0-7803-9546-8/06/\$20.00© 2006 IEEE

<sup>2</sup> IEEEAC paper #1177, Version 3, Updated Dec., 7 2005



**Figure 1.** Use of AAC for emergency communication in adverse conditions among airplanes as well as airplanes and ground stations

## 2. BACKGROUND

The simplest method for broadcast is flooding. Its advantage is its simplicity. However, for a single broadcast, flooding generates abundant retransmissions resulting in battery power and bandwidth waste. In addition, the retransmissions of close nodes are likely to happen at the same time. As a result, flooding quickly leads to message collisions and channel contention. This is known as the broadcast storm problem [1].

The broadcast problem has been extensively studied for multi-hop networks. Optimal solutions to compute Minimum Connected Domination Set (MCDS) [2] were obtained for the case when each node knows the topology of the entire network (centralized broadcast). The solutions presented in [2],[3],[4] are deterministic and guarantee a bounded delay on message delivery, but the requirement that each node must know the entire network topology is a strong condition, impractical to maintain in wireless networks.

In a counter-based scheme [1], a node does not retransmit if it overhears the same message from its neighbors for more than a prefixed number of times and in a distance-based scheme [1], a node discards its retransmission if it overhears a neighbor within a distance threshold retransmitting the same message. The key feature of these techniques is that they do not need neighbor knowledge.

Source Based Algorithm [5], Dominant Pruning [6], Multipoint Relaying [7], Ad Hoc Broadcast Protocol [8], Lightweight and Efficient Network-Wide Broadcast Protocol [9] utilize two-hop neighbor knowledge to reduce number of transmissions.

A good classification and comparison of most of the proposed protocols is presented in [10]. It is also concluded that Scalable Broadcast algorithm (SBA) [5] and Ad Hoc Broadcast Protocol (AHBP) [8] perform very well as the number of nodes in the network is increased. Both these techniques are based on two-hop neighbor knowledge, achieved via periodic *hello messages*.

Location Aided Broadcast [11] presents three location-aided broadcast protocols to improve communication overhead and summarizes the shortcomings of various protocols. In self-pruning methods [5],[12],[13], each node makes its local decision on forwarding status: forwarding or non-forwarding.

A review of MPR and DS based protocols for ad hoc networks is presented in [14]. A broadcast protocol is presented in [15] for regular grid-like sensor networks. In Gossip-based routing [16], a node probabilistically forwards a packet to control the spreading of the packet through the network; the probability typically being around 0.65. Although, this simple mechanism reduces the number of redundant transmissions, there is still a great scope for improvement. Probability based Broadcast Forwarding (PBBF) [17] extends Gossip based routing to sleeping sensor networks and attempts to trade off energy for latency.

Two key approaches to address the broadcast problem are Connected Dominating set (CDS) and multipoint relaying (MPR). Multipoint relay protocols belong to the family of neighbor-designating methods. In these schemes, the sending node selects neighboring nodes that should relay the message to complete the broadcast. The identities of the selected nodes are recorded in the retransmission packet as a forward list. Several schemes based on one or both of above approaches have been proposed in literature [6],[7],[8],[9],[12],[13]. All above protocols require either one-hop or two-hop neighbor knowledge. Every node can deduce its two-hop neighbor information, by having every node include its one-hop neighbor table in the hello message,. For highly mobile networks, the information in the neighbor tables becomes obsolete very fast, leading to routing failure or increase of hello messages frequency, which increases the chances for collision.

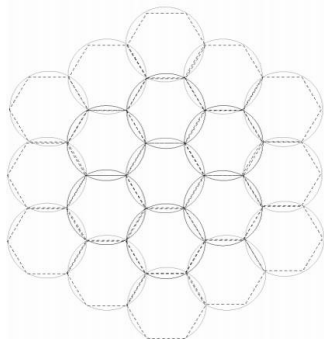
At best of our knowledge we are not aware of any other work related to broadcast protocols designed for communication among airplanes.

### 3. AIR TO AIR COMMUNICATION

The rationale behind our protocol is that to broadcast a packet over a network, we can select a few strategic nodes with the goal of minimizing the number of transmissions. The strategy to select such nodes was inspired by the Covering Problem presented in the following. We extend the ideas of our previous protocol BPS [18] to air communications.

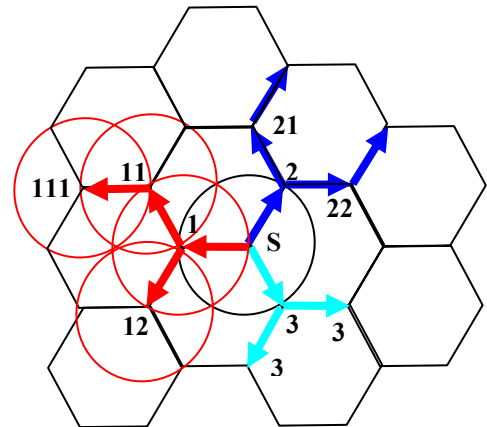
#### 2D Covering Problem

The Covering Problem can be stated as follows: "What is the minimum number of circles required to completely cover a given two-dimensional space?" Kershner [19] showed that no arrangement of circles could cover the plane more efficiently than the hexagonal lattice arrangement shown in Figure 2. Initially, the whole space is covered with regular hexagons, with sides  $R$  and then, circles are drawn to circumscribe them.

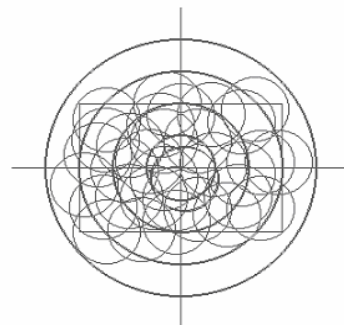


**Figure 2.** Covering Problem: What is the minimum number of circles required to completely cover a given two-dimensional space?

This problem can be modified as follows: "What is the minimum number of circles of radius  $R$  required to entirely cover a two-dimensional space with the condition that the center of each circle being placed lays on the circumference of at least one other circle." If  $R$  is the node's communication range, then this problem would be that of covering a given area with radio signal. We used this idea to develop our protocol for 2D case as shown in Figure 3 [18]. We have modified the covering problem to the following algorithm, initially explained for ideal conditions. The area to be covered with radio signal is portioned into hexagons. The communication range of nodes determines the hexagons' length of sides. The Source  $S$  is at the center of one of the hexagons. In an ideal network, all other transmission nodes are at the hexagons' vertices, as shown in Figure 3. We will call the vertices of the hexagons strategic locations. The broadcasted packets are propagated along the sides of the hexagons. Any active node located inside a hexagon is reachable from at least one of the vertex nodes of the hexagon. Of course, in real conditions, it is impractical to assume that active nodes are located at the hexagons' vertices. Thus, if the active neighbor nodes are not in the optimal strategy locations, the coverage figure will be distorted; moreover, the distortion effect may propagate as shown in Figure 4. A simple solution is to select the nearest active node to the supposed vertex.



**Figure 3.** Our Solution for the 2D Modified-Covering Problem shown in plane XY.

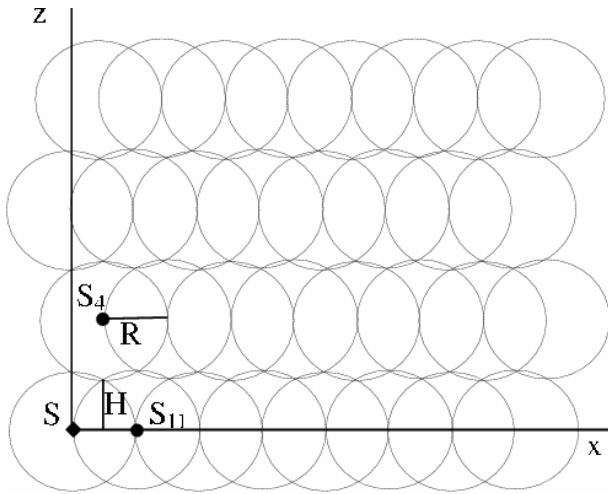


**Figure 4.** Our solution for the 2D Modified-Covering Problem in real conditions. As shown, in real conditions the retransmissions happen not at the hexagons' vertices.

### 3D Covering Problem

The aerospace network region can be represented as a 3-dimensional space. We observe that the height of the region might not be very large when compared to the radio transmission range. In such cases, we propose to divide the aerospace region into sub-regions of height  $H$  ( $H \leq 2R$ ) each and apply AAC in these sub-regions individually. The horizontal view in an ideal case is as shown in Figure 5. Notice that the overlap between two transmissions in two adjacent layers is very small and hence ignored. From simple geometry, it could be seen that  $H = \frac{\sqrt{3}}{2} R$  would

be an optimal value in an ideal case. But, at the same time due to distortion, it might be desirable to select  $H$  such that there is an overlap. At lower densities a lower value of  $H$  might be desirable. We studied the performance of AAC through simulations and have observed that at  $H = 0.8 \times R$ , we achieve the best performance for various densities. At this value, AAC achieved very high reachability and a further reduction of  $H$  does not increase considerably the reachability.



**Figure 5.** Air to Air Communication Protocol for 3D shown in plane XZ

#### Algorithm

It should also be observed that a node could receive a packet more than once - from different directions and from different nodes, each node specifying different optimal strategic location (because of distortion). This may cause two nodes very close to each other to retransmit. We propose to avoid these transmissions by having a node keep track of its distance  $d_m$  to the nearest node that has retransmitted the packet and to have a node retransmit only when its distance to the nearest transmitting node is greater than a threshold  $Th$ .

After the source  $S$  sends the packet, the first round of selected nodes are the three neighbor edge nodes 1, 2, 3, in

horizontal plane  $XY$ , as shown in Figure 3. Then the selection in the horizontal plane  $XY$  continues with two neighbor edge nodes, for example for the node 11, in the next round nodes 111 and 112 are selected (as shown in Figure 3). The two selected nodes make a  $\pm 120^\circ$  angle from the node from which the message was received. In this way, the messages are propagated until they cover the entire network horizontal area. In case we need to broadcast in other horizontal planes, besides the one where the source  $S$  ( $0, 0, 0$ ) is located, then  $S$  sends a packet designated for the upper plane and more precisely for the closest node  $S_4$  to the location  $(R/2, 0, H+R)$ , shown in Figure 5.

In the following we describe the AAC algorithm for a given  $XY$  plane. All  $XY$  planes follow the same algorithm; only the coordinate  $Z$  changes accordingly.

Each broadcast packet contains two location fields,  $L_1$  and  $L_2$  in its header. Whenever a node transmits a broadcast packet, it sets  $L_1$  to the location of the node from which it received the packet and sets  $L_2$  to its own location.

The Source Node  $S$  sets both  $L_1$  and  $L_2$  to its location  $(S_x, S_y, S_z)$  and transmits the packet.

1. Upon the reception of a broadcast packet, a node  $M$  discards the packet if  $M$  has transmitted the packet earlier, or if a node which is very close has already transmitted this packet, i.e., if  $d_m < Th$ .
2. If the packet is not discarded,  $M$  finds the nearest vertex  $V$  (for example node 1 in Figure 3) of a hexagon with  $(S_x, S_y, S_z)$  as its center coordinates and with  $(S_{(x+R)}, S_y, S_z)$  as one of its vertices. It computes its distance  $l$  from  $V$  and then delays the packet rebroadcast by a delay  $d = l \times R$ .
3. After delay  $d$  elapses,  $M$  again determines if it has received the same packet again and if the packet can be discarded (for the same reasons mentioned above). Thus, delaying enables a node to decide if it is the nearest node to the strategic location. If the packet cannot be discarded,  $M$  updates  $L_1$  to location of the node from which it received the packet and  $L_2$  to its location, sets  $d_m$  to zero and transmits.

Choosing low delay values decrease the time needed to broadcast a message all over the network, while high delay values help reduce redundant transmissions in instances where two nodes are of comparable distance from the strategic location. The delay function we used causes a packet to be delayed a maximum of 50 ms per retransmission, though typically this value lies around 10 ms. In dense networks, the delay values are much less than 10 ms.

The purpose of having the threshold  $Th$  is to prevent two active nodes that are very close to each other from transmitting, thus reducing the redundancy. The key factors depending on  $Th$  are the number of transmissions and the delivery ratio. As  $Th$  increases, the number of transmissions decreases. This happens because when  $Th$  increases, the minimum distance between any two transmitting nodes increases. This in turn implies that additional area covered increases, and hence, the number of transmissions needed for covering entire network decreases. The higher the number of transmissions, the higher is the redundancy, and therefore the greater is the probability that a node receives broadcast.

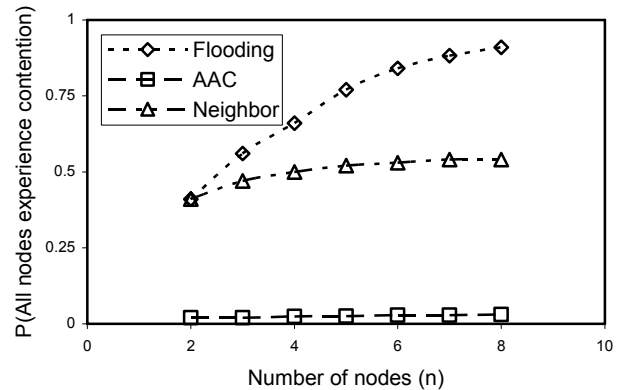
Therefore, for higher delivery ratios, lower  $Th$  is preferred. Through extensive simulations we have found that for a threshold value of  $Th = 0.35 \times R$ , a delivery ratio of around 98% is achieved and for  $Th = 0.4 \times R$ , the delivery ratio is close to 95%. However, when  $Th = 0.45 \times R$ , the delivery ratio falls to around 90%. This is understandable, because with the increase in threshold value, the number of retransmitting nodes decreases. For all further simulations, we use threshold value of  $Th = 0.35 \times R$ .

#### 4. SIMULATIONS

We have used ns-2 simulator [20] to evaluate the performance of our protocol. We considered an aerospace region of  $10000m \times 10000m \times 2500m$  with varying number of nodes. Every simulation is repeated until the 95% confidence intervals of all average results are within  $\pm 5\%$ . The simulations are aimed at studying the performance of AAC in different networks. We first study the effect of AAC in reducing the contention and compare it with flooding and neighbor selection protocols. Then, we concentrated on the algorithm efficiency by studying the performance of AAC in highly mobile networks.

##### Contentions

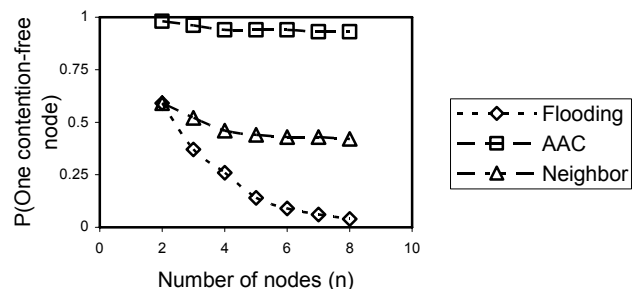
We analyze the performance of various protocols in terms of contention caused by the protocol. To address the contention problem, consider the situation where a node  $i$  broadcasts a message and there are  $n$  nodes hearing this message. If all these nodes try to rebroadcast the message, contention may occur because two or more nodes are likely to be close and thus contend with each other on the wireless medium. We studied the probability of contention through simulations by randomly placing  $n$  nodes in node  $i$ 's transmission range. We observed the probability that all  $n$  nodes experience contention and probability of having one contention-free node. The results are shown in Figures 6 and 7. We considered flooding, protocols in which the broadcasting nodes proactively chose neighbors to rebroadcast (neighbor) [6],[7],[8],[9],[12],[13] and AAC.



**Figure 6.** The probability that all  $n$  nodes experience contention

From Figure 6, we can see that with flooding, the probability that all  $n$  nodes experience contention increases rapidly and is more than 0.8 even in presence of just six neighbors. For neighbor protocols, the probability increases at a much slower pace because number of retransmitting nodes (depends on network topology) might not increase with the increase of number of neighbors. With AAC, the probability is almost zero. This is essentially because of two reasons - first, in AAC at most three neighbors would be retransmitting irrespective of number of neighbors; second, because of delay based self-selection of retransmitting nodes, the probability that two nodes experience the same delay is very low, thus reducing further the probability of collision.

The probability of having one contention-free node is shown in Figure 7. Understandably, with flooding this probability drops sharply as  $n$  increases. Further it is more unlikely to have more contention-free hosts. With neighbor protocols, because only few nodes retransmit, the probability does not decrease as rapidly as with flooding. Again for the same reasons mentioned above, with AAC, the probability remains close to one.

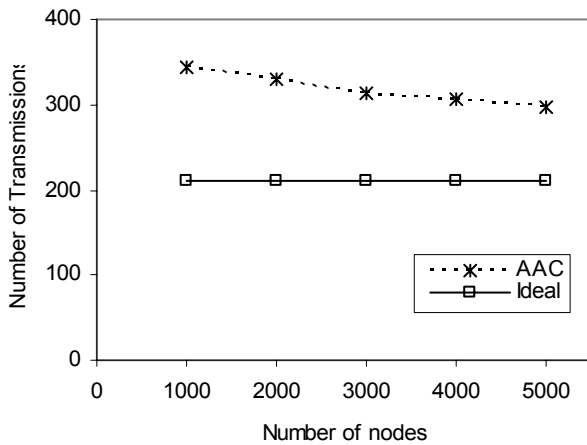


**Figure 7.** The probability of having one contention-free node

### Efficiency

The purpose of this experiment is to evaluate the performance of AAC in networks of different sizes and different densities. We include a "best-case" bound provided by the simulation results in ideal case scenarios. It is impossible for any algorithm to perform better than the performance in ideal case scenario and unlikely to perform worse than simple flooding. Thus, these two bounds provide a useful spectrum to gauge the performance of our protocol. For this study, we varied the number of nodes from 1000 to 5000.

Figure 8 presents the performance of AAC. It can be seen that AAC scales with the number of nodes. In fact, as the number of nodes increases, the performance is better. This is because of the fact that at higher densities, the probability that a node close to the ideal/strategic location is found is higher and hence a closer approximation to the ideal case is achieved. In presence of 1000 nodes, over 35% of nodes retransmit while in presence of 5000 nodes just over 6% of nodes retransmit.



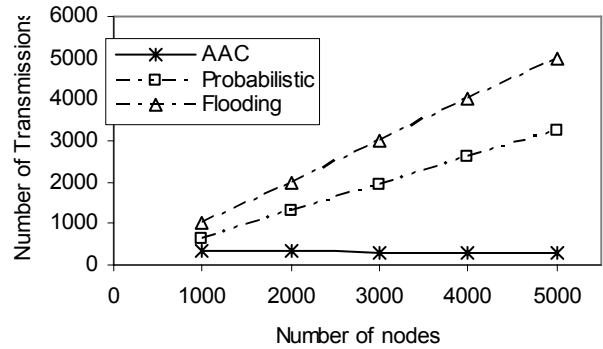
**Figure 8.** Number of retransmissions of Air to Air Communication Protocol vs. Density

Figure 9 compares the performance with probabilistic flooding [16] and flooding. In probabilistic flooding each node retransmits with a probability of 0.65. As noted, AAC outperforms both the protocols.

### Mobility

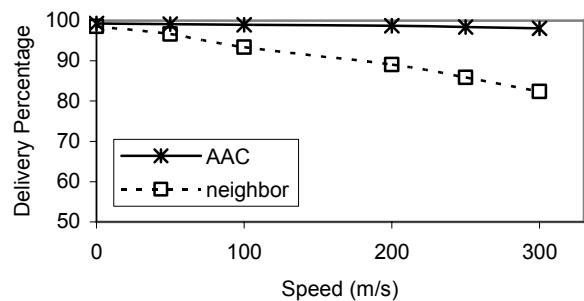
This section presents the simulation results of AAC and neighbor knowledge based protocols depending on the mobility of nodes. Airplanes move at very high (supersonic) speed. Therefore, the performance of protocols should be evaluated in such realistic scenarios as shown in Figure 10. The performance of AAC remains unaffected, as AAC algorithm uses minimal neighborhood information.

But, the performance of neighbor based protocols rapidly deteriorates with increase in speed and its performance is also affected by the *hello interval*.



**Figure 9.** Comparison among Air to Air Communication Protocol, flooding and probabilistic flooding

The neighbor knowledge based protocols use *hello messages* to gather the neighborhood information. With a *hello interval* of  $t$  seconds, the neighbor information (that is obtained through the hello messages of neighbors) would always be outdated by an average of  $t$  seconds. For instance, if  $t = 10$  seconds and a nodes speed is 300 m/s, then the node would have moved up to 3000m before its information has been conveyed to one of its neighbors. Also, once a node gets this information, it is not updated again until 10 sec later. Thus, a node could have moved up to 6000m before its information is updated at its neighbors. Also, the average time by which a node's information at neighbor based protocol is out-dated is 15 seconds ( $t + (0 + t)/2$ ), which corresponds to a displacement up to 4500 m. This shows the intensity of the effect the mobility has on these protocols. Thus, the *hello interval*  $t$  should be very small for efficient performance of neighbor knowledge based protocols, which in turn means that the bandwidth overhead due to *hello messages* is very high.



**Figure 10.** Delivery percentage vs. node speed for ACC and neighbor knowledge based protocol

### Average Delay per Hop

We observed that the average delay a node has to wait at each hop before retransmitting and the results are presented in the Table 1. We consider 50 ms as maximum allowed delay per hop. We observe that even at low densities, the delay is around 17 ms, while at high densities the delay is very low and nearly negligible.

In Table 1, we show the delay per hop depending on node density.

Table 1. Delay per hop vs. Density

Density	Delay per hop [ms]
4	16.9
6	14.1
16	8.4
25	7.2
100	3.7

## 5. CONCLUSIONS

We presented Air to Air Communication Protocol, a broadcast protocol that enables efficient transmission to all nodes in an ad hoc network of airplanes. Because of its geometric nature, AAC minimizes the number of retransmissions while maintaining high reachability. AAC's communication overhead is negligible because no hello messages are used. AAC performs very well in the highly dynamic networks such as the ones composed by fast moving nodes-airplanes. AAC can enhance considerably the communication capability in case of emergency situations due to coordinated attacks.

## 6. ACKNOWLEDGEMENTS

The research was supported in part by the NSF under Grant #0413187.

## REFERENCES

- [1] S. Y. Ni et al., "The Broadcast Storm Problem in a Mobile Ad Hoc Network," *ACM MOBICOM*, August 1999, pp. 151-162.
- [2] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," In *Proceedings of European Symposium on Algorithms (ESA)*, 1996.
- [3] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg, "A lower bound for radio broadcast," *J. Comput. Syst. Sci.*, vol. 43, Oct. 1991, pp. 290-298.
- [4] I. Gaber and Y. Mansour, "Broadcast in radio networks," In *Proceedings of 6th Annu. ACM-SIAM Symp. Discrete Algorithms*, San Francisco, CA, Jan. 1995, pp. 577-585.
- [5] W. Peng and X. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks," In *Proceedings of MOBIHOC*, 2000.
- [6] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks," In *Proceedings of the ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, 2000.
- [7] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying: An efficient technique for flooding in mobile wireless networks", *Technical Report 3898*, INRIA - Rapport de recherche, 2000.
- [8] W. Peng and X. Lu, "AHBP: An efficient broadcast protocol for mobile ad hoc networks," *Journal of Science and Technology*, Beijing, China, 2002.
- [9] J. Sucec and I. Marsic, "An efficient distributed network-wide broadcast algorithm for mobile ad hoc networks," *CAIP Technical Report 248*, Rutgers University, September 2000.
- [10] Brad Williams, Tracy Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," In *Proceedings of the third ACM international symposium on Mobile ad hoc networking and computing*, June 2002
- [11] M. Sun and T. H. Lai, "Location Aided Broadcast in Wireless Ad Hoc Network Systems," In *Proceedings of IEEE WCNC 2002*, March 2002, pp. 597-602.
- [12] J. Wu and H. Li, "On calculating connected dominating sets for efficient routing in ad hoc wireless networks", In *Proceedings of the International Workshop on Discrete Algorithms and methods for Mobile Computing and Communication (DIAL-M)*, 1999, pp. 7-14.

- [13] Jie Wu and Fei Dai, "Broadcasting in Ad Hoc Networks Based on Self-Pruning," In *Proceedings of IEEE INFOCOM 2003*, San Francisco, CA, 2003.
- [14] Gang Lu, Narayanan Sadagopan, Bhaskar Krishnamachari, Ashish Goel, "Delay Efficient Sleep Scheduling in Wireless Sensor Networks," In *Proceedings of IEEE INFOCOM 2005*, Miami, FL, March 2005.
- [15] Shlomi Dolev, Ted Herman, and Limor Lahiani, Ben-Gurion, "Polygonal Broadcast for Sensor Networks," *2nd IEEE Upstate New York Workshop on Sensor Networks*, 2003.
- [16] Z. Haas, H. Li., "Gossip based Ad Hoc Routing," In *Proceedings of IEEE INFOCOM 2002*, June 2002.
- [17] I. Stojmenovic, J. Wu, "Broadcasting and activity scheduling in adhoc networks," *Mobile Ad Hoc Networking*, IEEE/Wiley, 2004, pp. 205-229.
- [18] A. Durreesi, V. Paruchuri, S.S. Sitharama, R. Kannan, "Optimal Broadcast Protocol for Sensor Networks," *IEEE Transaction on Computers*, Vol. 54, Issue 8, August 2005, pp. 1013-1024.
- [19] R. Kershner, "The Number of Circles Covering a Set," *Amer. J. Math*, 61, 665-671, 1939.
- [20] ns-2. Network Simulator <http://www.isi.edu/nsnam/ns/>

## BIOGRAPHY

Arjan Durreesi is an Assistant Professor at Louisiana State University, Baton Rouge, LA. His research interests include network architectures, ad hoc, wireless networking, Traffic Management, QoS, security. He is the author of five book chapters and over 100 articles in in Journals and International Conference proceedings. Dr. Durreesi obtained an M.S. degree and Ph.D. (all summa cum laude) in Electronic-Telecommunications, in 1991 and 1993, respectively and a Diploma of Superior Specialization in telecommunications from La Sapienza University in Rome, Italy and Italian Telecommunications Institute in 1991. He is an area editor for the *Ad Hoc Networks Journal*. He was Program Chair of IEEE AINA2006, Co-Chair of the First and Second International Workshops in Heterogeneous Wireless Sensor Networks HWISE2005-06, Program Vice Chair of AINA2004 and ICPADS 2005, and Program Area Chair of AINA2005. He received the appreciation certificate from IEEE Computer Society in 2005. He is a senior member of IEEE, member of ACM and SPIE.



Vamsi Paruchuri received the BTech degree in electronics and communications engineering from Sri Venkateswara University, Tirupati, India, in 2001 and the MS degree in electrical engineering from the Ohio State University, Columbus OH, USA, in 2003. He is currently a PhD candidate in computer science at the Louisiana State University, Baton Rouge LA, USA. His research interests include routing and security protocol design, analysis, and implementation for wireless networks. He is a student member of the IEEE.



Leonard Barolli received B.E. and Ph.D. degrees from Tirana University and Yamagata University in 1989 and 1997, respectively. Presently, he is a Professor at the Department of Information and Communication Engineering, Fukuoka Institute of Technology (FIT). Dr. Barolli has published more than 100 papers in Journals and International Conference proceedings. He was an Editor of the *IPSJ Journal* and has served as an Guest Editor for many Special Issues of International Journals. Dr. Barolli has been a PC Member of many International Conferences. He was a PC Co-Chair of AINA-2003, PC Chair of AINA-2004. He is a PC Chair of ICPADS-2005 and General Co-Chair of AINA-2006. His research interests include network traffic control, fuzzy control, genetic algorithms, agent-based systems, ad-hoc networks, sensor networks and P2P systems. He is a member of SOFT, IPSJ, IEEE Computer Society and IEEE.



Raj Jain is a Professor of Computer Science and Engineering at Washington University in St. Louis. He has been also a Co-founder and Chief Technology Officer of Nayna Networks, Inc. He was a Professor of Computer and Information Sciences at The Ohio State University in Columbus until August 2002 and then an Adjunct Professor until August 2004. He is a Fellow of IEEE, a Fellow of ACM and is on the Editorial Boards of *Computer Communications*, *Journal of High Speed Networks*, *Mobile Networks and Nomadic Applications*, *International Journal of Virtual Technology and Multimedia* and *International Journal of Wireless and Optical Communications*. He has 14 patents, more than 40 journal and magazine papers, and more than 60 conference papers. His papers have been widely referenced and he is known for his research on congestion control and avoidance, traffic modeling, performance analysis, and error analysis.

