Proceedings of the Med-Hoc Net 2003 Workshop
Mahdia, Tunisia 25-27 June, 2003

SYSTEM SUPPORT FOR SMALL-SCALE AUCTIONS

Nan Lin and Santosh Shrivastava

School of Computing Science
University of Newcastle upon Tyne
Newcastle upon Tyne U.K. NE1 7RU

email: nan.lin, santosh.shrivastava@ncl.ac.uk

ABSTRACT

Mobile wireless computing devices provide interesting opportunities for building new styles of interactive applications by exploiting ad hoc networks. The basic networking technology needed to support such applications is readily available. The paper describes the design and implementation of a small scale auction application in an ad hoc network. It is important to develop a variety of interactive applications using ad hoc networks in order to understand what functionality can suitably be incorporated in the mobile computing middleware. In the case of this particular auction application, the paper describes how an application specific ad hoc network can be formed and maintained in an economical manner.

1 INTRODUCTION

With the development of wireless communication and mobile computing, new ways for people to interact with each other and their surrounding environment are emerging. Mobile devices, such as Personal Digital Assistants (PDAs) with wireless communication interfaces are able to communicate directly with each other if they are 'close enough'. If such devices are also able to act as message relays (routers), then a very powerful facility in the form of a mobile ad hoc network can be made available to applications.

We speculate that in future people carrying PDAs (and similar devices, such as mobile phones etc.) will be able to form a Spontaneous Information System (SIS) [1]. The system is spontaneously created as soon as, at least, two mobile entities approach each other. Afterwards the entities have the ability to establish communication, and exchange implicitly or explicitly some information. Ad hoc networks can be used for supporting a variety of SIS applications. In this paper we describe the development of one such application: an auction in a small-scale ad hoc network. It could happen in a limited space like a flea-market, where people carrying PDAs form an ad hoc network. The system is created spontaneously as soon as, at least two people approach each other. Auction information is disseminated in the system and interested bidders try to submit their bids to win the auction. The whole process may last for a small duration, during which communication quality could be variable. Our objective is to investigate how fair mechanisms to conduct an auction can be developed. We believe that it is important to develop a variety of interactive applications using ad hoc networks in order to understand what functionality can suitably be incorporated in the mobile computing middleware. In the case of this particular auction application, we describe how an application specific ad hoc network can be formed and maintained in an economical manner.

The paper is organized as follows: section 2 we introduce the scenario, requirement and procedures of the application; section 3 gives the system design; section 4 reports test results; section 5 reviews the related work and section 6 concludes the paper.

2 AN AUCTION IN A SMALL AD HOC NETWORK

2.1 Application Scenario

One of the big challenges in on-line auctions is *fairness* which means that all participant bidders in the auction have an equally fair chance of submitting a successful bid, and all sellers have an equally fair chance of selling their items at a price that reflects market demand. All other general requirements for auctions must be also respected, whatever the auction method being chosen. Currently available on-line auctions essentially meet the fairness criterion by prolonging the bidding procedure. However, if the bidding time is reduced to form short duration, 'real-time' auctions, then maintaining fairness in the presence of variable communication delays becomes difficult [2].

In the context of auctions and ad hoc networks, an attractive computing environment emerges: when people who carry PDAs with communication interface are close together, their devices could automatically explore surroundings and directly establish communication, form an ad hoc network and exchange some information implicitly or explicitly. Afterwards a spontaneous auction could be created. Sellers push auction information onto their network neighbours and those who receive messages relay them until everyone knows what is happening. Interested buyers then start their bidding. It is an instance of open cry auction, defined by the mode of communication in which all status information is conveyed immediately and globally to all participants.

This scenario may happen in a club. A group of football fans carrying PDAs always meet there for drinking, chatting and betting before kicking off. Some one has a spare ticket and wants to sell it. Some others who have no tickets would like to get one. If the PDAs are turned on (maybe some in standby mode), a seller's broadcast message should reach every PDA in the network. Interested buyers submit their bidding within a short duration; the process advances round by round till the highest bidder wins the auction.

2.2 Application Assumptions

For the sake of simplicity, we assume that only one auction is in progress. The following additional assumptions have been made:

1. The nodes can be far enough apart so that not all of them are within the range of each other.
2. The nodes may be mobile so that two nodes within range at one point in time may be out of range moments later.
3. The nodes are able to assist each other in the process of relaying packets.
4. The wireless network communication between any two nodes works well in both directions.
5. At the application level, bidders communicate only with auctioneers, they don't communicate with each other. (Of course, at the lower level, where the ad-hoc network is formed and maintained, communication between bidders may well take place for message relaying, but this is not visible at the application level.)
6. The auction happens in some limited geographical place (say, no more than four hops required) and the number of bidders is relatively small (say less than a hundred).
7. All nodes are in the same network, so they share the same network name.

2.3 Application Procedure

We assume that the auction application is running on each PDA and monitoring events of interest in the network. Then the following sequence of actions takes place:

1. Forming a virtual auction house. A seller (auctioneer) node broadcasts auction information to the network. Interested nodes send registration information to the seller. The auctioneer builds up a bidder list.
2. Rules announcement. The auctioneer broadcasts the details of the auction. Bidders read carefully and prepare for bidding.
3. Start of bidding. Bidders submit their bids. The auctioneer notifies the latest information promptly to all the bidders; they respond in a short time, overbid, wait or quit.
4. Closing the auction. Auction closes according to the rules. The auctioneer announces the result to the bidders and communicates with the winner directly. They conclude the deal when they meet face to face.

3 SYSTEM DESIGN

3.1 Overview

There are two modes of communication required by the application: broadcast from the auctioneer to bidders and unicast from a bidder to the auctioneer. The key design problem is how to ensure reliable communication between the two parties who may well be several hops away from each other. Our solution is based on the following simple ideas. Broadcast is implemented by flooding (diffusion): a node receiving a broadcast message diffuses it by re-broadcasting; the node re-broadcasting a message also adds its identity in the path field kept in the header part of the message. A node receiving a broadcast message can find out the path taken by that message from the information in the header, and uses that path for unicasting any response message to the auctioneer; a node receiving such a message forwards (relays) it to the next node on the path.

Our protocol (presented in the next subsection) is a simplified version of the dynamic source routing (DSR) protocol [3]. These simplifications are possible because of the nature of the application, and are discussed in a subsequent part of the paper. An effective design should not only cope with the general problems in ad hoc networks such as mobility of users but also the "broadcast storm" problem [4,5]. Our approach to these problems is also discussed and evaluated in the paper.

3.2 Protocol Design

Bidders and the auctioneer exchange data through formatted packets (see the section 3.5). Every bidder maintains a pair of variables for recording the auctioneer address and highest sequence number received so far (say S_h). Bidders discard any packets with sequence numbers less than S_h. Otherwise bidders append their own address to the source path kept in the control part of a packet, update S_h and re-broadcast to the network. This procedure is illustrated in Figure 1. (Tables in Figure 1 show the simplified structure of packets: row 1 only records sequence number, row 2 records the source route and row 3 shows payload data.) A bidder learns its source route to the auctioneer by reversing the source route in the control part of the packet; it stores this path information in its route table. Bidders update their route tables whenever they receive a new packet from the auctioneer containing a different source route. Bidders use their source routes when they unicast bid messages to the auctioneer. Every bidder keeps only one source route to the auctioneer in order to minimise its memory requirements. Due to node mobility in the network, the source route of a bidder sometimes will not work. This could happen if a bidder suddenly moves out of its neighbour's range just after the bidder's PDA updates its route table. The result is that the auctioneer will not receive any packets from this bidder. A simple way of minimising the risk of this happening is for the auctioneer to broadcast periodically its current information. This way, a bidder may be able to reconnect to the auctioneer, and resume bidding.

Straightforward flooding without any control mechanisms is simple and effective for small networks. But when density of nodes becomes higher, far too many redundant packets will be re-broadcast leading to the broadcast storm [4]. In order to alleviate this problem, we adopted and implemented the counter-based flooding scheme proposed in [4,5]. This scheme maintains an inverse relationship between the number of times of a redundant packet is received at a node within a short interval and the probability of that node, on re-broadcasting, being able to reach an area not yet covered. There are two constants involved: maximum waiting interval (Tmax) and maximum number of redundant packets received (Cmax). Upon receiving of a previously unseen packet, the bidder initialises a counter with a value of zero and sets a random waiting time (RWT), which is between 0 and Tmax. During RWT, the counter is incremented by one for each redundant packet received. After RWT expires, if the counter value is less than Cmax, the redundant packet is re-broadcast. Otherwise the packet is discarded.

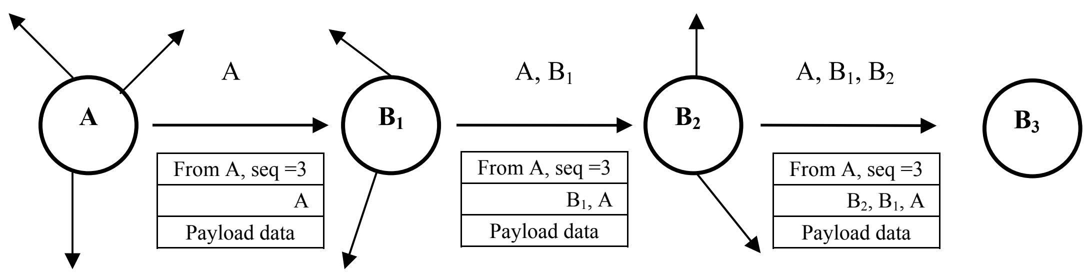

Figure 1: Source path building in a packet from the auctioneer A

3.3 Implementation

There is much activity on developing routing protocols for ad hoc networks. However, most of proposed routing algorithms are still being researched, with the result that PDAs currently support the traditional TCP/IP and UDP/IP. We have therefore implemented our own ad hoc network routing protocol described earlier, using Java and UDP. The architecture of our application consists of the following components: graphic user interface (GUI), data-processing module and routing function module. (see Figure 2).

GUI traps the input from users as well as displaying messages from the network. Data-processing module is responsible for dealing with data from the GUI or the network socket. Routing function sub-module maintains source routing path.

3.4 Packet Structure

Fixed length UDP packets (512 bytes) transferred in ad hoc networks are made up of two parts: control part (first 64 bytes) and real payload data, see Table 1. (At present, only one auction is supported by our application but this could be extended to support many auctions at the same venue.)

3.5 Fairness Guarantee

In our application, the "fairness" problem comes mainly due to variable communication delays, bearing in mind packets from bidders who have more hops would reach the auctioneer slowly. If the auctioneer broadcasts the latest bid information immediately when it receives every new bid, the bidders nearest to the auctioneer would gain an advantage over those that are multi-hops away. Another problem is that it causes extra network traffic. For instance if there are four bids coming one by one, the auctioneer may broadcast four times. In our design, we introduce a short interval into the network layer of the auctioneer. When the network layer receives first new bid, it does not immediately deliver to the application layer but starts a timer waiting for other possible bids. After the timer expires, the network layer delivers all the received information to the application layer. The application layer decides what the highest bid is and broadcasts the next bid information. So form the view of the application layer, all the biding information comes at the same time. It ensures the fairness. We suggest that the short interval is great then $(M_{hop}-M_{actual}) * 2 * T_D$. M_{Hop} is the maximal number of hops. M_{actual} is the actual hops number of first coming bid packet and T_D is average delay for one hop. In practice, the average network delay for one hop is of the order of 20 to 73 milliseconds (See Section 4) and bidding decisions are done by people and not software. We assume that people would respond to the latest bidding information variably from a few seconds to one minute. So the first bid can come from any of the bidders. To simplify the design, we set the short interval to a fixed value of one second.

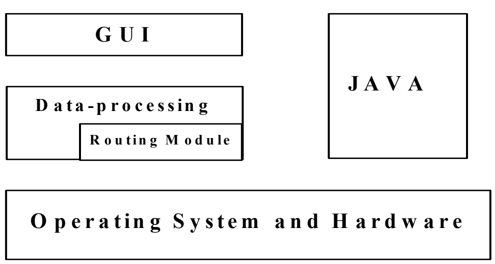

Figure 2: Application architecture

Table 1: Packet structure

Index (Bytes)	Description
0~3	Sequence number
4	Packet type. See note*.
5	Total hops number
6	Current hops number
7	Auction ID
8~27	Source route
28~63	Reserve
64~511	Payload data.

* 0: auctioneer to bidders, management information
1: auctioneer to bidders, announce the latest price
2: bidders to auctioneer, for enquiries
3: bidders to auctioneer, submit bidding

4 TESTING

4.1 Testing Environment

In order to test the application we conducted our experiment using a laptop (IBM Thinkpad) as a data collector (auctioneer) and five PDAs (HP Jornada 720 running Windows for Handheld PC and HP Chai Java virtual machine) as bidders. Wireless Ethernet 802.11b standard is used and the speed set to 2Mbps on a channel (products are from two companies: Cisco Airnet 340 and Orinoco). Maximum hops limit is set to four. We do not set the timer in the network layer work. Three performance metrics are of interest:

1. Packet loss: the number of broadcast packets that are not received divided by the total number of sent packets;
2. Average latency: the interval from the time the broadcast was initiated to the time the last host finished its re-broadcast;
3. Saved ratio: (r-t)/r, where r is the number of PDAs receiving packets and t is the number of PDAs actually re-broadcasting packets.

Data collector sends 500 broadcast packets periodically (ten packets per second), records the time when it subsequently receives the re-broadcast packets from other PDAs. The same test repeats 5 times.

4.2 Simple Flooding Testing

Our first goal examines the efficiency of simple flooding (fig. 3(A)) in a system with one bidder node, two nodes and five nodes. There is no contention in the network with one bidder so it is the ideal situation for communication. The results are shown in Table 2. We should note that maximum latency is eight times than the average, though there are few packets who experience such long delays.

Table 2: One PDA with simple flooding

Maximum latency	160 ms
Average latency	20.7 ms
Packet loss	< 0.1%
Packet number whose latency is over 40 ms	< 3

Table 3 shows the result with two PDAs. There is slight contention in the network under this situation.

Table 3: Two PDAs with simple flooding

Maximum latency	165 ms
Average latency	22.7ms
Packet loss	< 0.1%
Packet number whose latency is over 40 ms	6.5%

Table 4 shows the result with five PDAs. There is fairly heavy contention in the network.

Table 4: Five PDAs with simple flooding

Maximum latency	300 ms
Average latency	73 ms
Packet loss	2%
Packet number whose latency is over 100 ms	17%

These results demonstrate that the “broadcast storm” problem exists in a shared radio network: when node density in a network grows, contention for controlling the channel become higher and without the help of RTS/CTS, timing of broadcasts is highly correlated; consequently leading to much longer average latency and higher lost rate.

4.3 Counter-based Flooding Test

We did these tests with five PDAs, set the value C_{max} to three and T_{max} to 25ms, 50 ms and 100 ms respectively, table 5 to 7 shows the result.

Table 5: Five PDAs with counter-based flooding
$T_{max} =$ 25 ms

Maximum latency	240
Average latency	46.8
Packet loss	0
Saved ratio	32.1 %
Packet number whose latency is over 100 ms	1.5%

Table 6: Five PDAs with counter-based flooding
$T_{max} =$ 50 ms

Maximum latency	250
Average latency	57.4
Packet loss	0
Saved ratio	36.7%
Packet number whose latency is over 100 ms	6.9%

Table7: Five PDAs with counter-based flooding
$T_{max} =$ 100 ms

Max latency	270
Average latency	75.7
Packet loss	0
Saved ratio	46.2 %
Packet number whose latency is over 100 ms	8.2%

It proves that performance of counter-based flooding is better than simple flooding: it reduces the number of redundant packets dramatically and the average latency is improved.

4.4 Mobility Test

The laptop and four PDAs are put in a house and divided into two groups which are physically separated by around 8 meters. A man carrying a PDA using simple flooding method outside of the house is walking. This PDA is the data initiator and collector. We assume that the initiator can reach the farthest computer via two hops so the latency record on the initiator is for two hops and four hops (see fig. 3(B). The results sometimes are poor: packet loss is over 30%. We found that the reason was when the initiator was on the border between one hop and two hops of farthest PDA. This PDA first learns a one-hop source route that then suddenly fails as the initiator moves out of direct range. If the initiator keeps walking in the range of two hops from farthest PDA, average latency for four hops is 57.8 ms, maximum delay is about 210 ms and packet loss is almost zero. A simple way of minimizing the risk of this happening is for the auctioneer to broadcast periodically its current information. This way, a bidder may be able to reconnect to the auctioneer, and resume bidding.

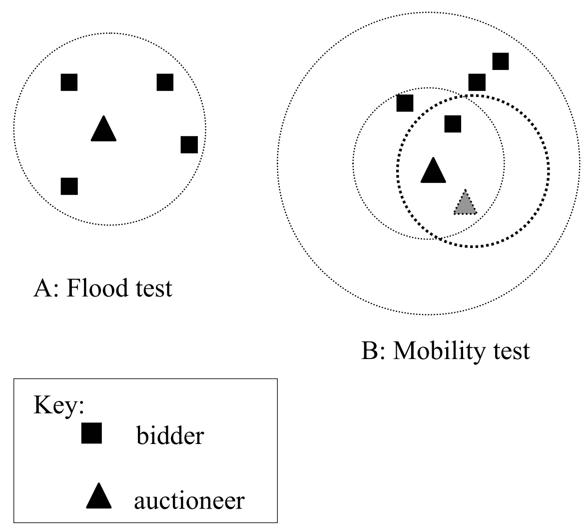

Figure 3: Testing scenarios

5 RELATED WORK

On-line Internet auctions have been researched extensively. [18] describes an application for auctioning goods on the Internet. Different commonly used auction mechanisms that are supported by the application, security requirements, pre-auction and post-auction interactions needed to complete auction are discussed. It presents a software prototype and introduces the various processes that comprise the auction. Finally it assesses how the delay, security and collusion aspects will need to be addressed in Internet based auctions. [2] discusses several temporal issues that arise in the context of auctions and argues that it is generally advantageous to design for maximal asynchrony and robustness to network delays. The difficulty of implementation of "real-time" auctions is not mainly technological, requiring identifying the group of participants and multicasting all bid information, but a matter of flexibility. Flexibility means that people would prefer to attend to auctions at their own convenience, rather than tune in at a designated uniform time. It also points that for auctions designed to be used by software agents, the temporal flexibility issue may be ameliorated somewhat. [19] presents an Internet auction system designed to investigate the real-time requirements. The delivery of bids is handled by a server agent residing on the user site. The system guarantees the fairness among different bidders by adjusting the expected delivery time to each bidder taking the individual network delay into consideration. [20] defines the concept fairness with respect of QoS requirements of responsiveness, accessibility, security and scalability. It presents a novel software architecture supporting Internet auctions. The goals have been achieved by replicating the auction service across a number of auction servers.

Currently the most fundamental research issue in mobile ad hoc networks is packet routing. Many protocols have been proposed, such as DSR [3], AODV [6], DSDV [7], TORA [8] and ZRP [9]. Much of the work is theoretical, with performance comparison of different protocols and test based on simulation study [10], [11], [12].

As stated earlier, our protocol is a simplified version of the DSR [3]. We discuss these simplifications. DSR is one of the most comprehensively studied on-demand routing protocols in ad hoc networks. Its key feature is the use of source routing: the sender knows the complete path to the destination en route. Hosts maintain a route cache storing such routing information. Data packets propagating in the network carry the source route in their header. When a host wants to send a packet to the destination for which it does not have the route, it sends a *route discovery* packet to dynamically find the route. Route request is flooded into the network. Each host receiving this kind of packets rebroadcasts it, unless it is the destination or it has a route in its cache. Such a host replies to the request with a *route reply* packet that is routed back to and is stored in the initial source. If any link on a source route is broken, a *route error* packet is generated and sent back to the source using the part of the route traversed so far, erasing all entries in its route cache along the way that contain the broken link. A new route discovery must be initiated by the source, if the route is needed and no alternate route is available in the cache. Several optimizations to this basic protocol have also been proposed in its specification.

We are able to eliminate several of the DSR features and come up with a very economical version by exploiting the nature of our application. In particular, we do not need to exchange route discovery, route reply and route error packets. Bidders update their routes when they receive the latest auctioneer packets. Since it is in the interest of the auctioneer to reach all the bidders, we ensure that the auctioneer remains lively by periodically broadcasting its current information (even during phases it is expecting bids). This is achieved transparently to the application. Although this does not guarantee that a bidder will never have a broken path, it does minimise the risk of lost bids.

In practice broadcast is a common operation in many ad hoc network applications, sometimes it is the only way to exchange information among members due to host mobility. In fact most of the routing protocols use flooding for broadcasting when a node tries to set up a path to an unknown node. Flooding leads to the “broadcast storm” problem resulting in inefficient use of node and network resources [4]. Several broadcasting techniques whose goal is to minimize the number of re-broadcasts while attempting to ensure that broadcast packets are delivered to each node in the network have been proposed. They can be categorized as: probability-based (counter-based flooding belongs to this category) [4], area-based [4] and neighbour-knowledge-based [13], [14], [15]. Simulation studies [4], [5], [14], [15] pinpoint the efficiencies and deficiencies in these protocols and propose solutions to correct for specific problems.

Not much work on interactive applications has been reported. However it is important to reconcile simulation study with real-world test of broadcasting, as simulations may not consider all the factors that may extraordinarily affect broadcast performance. [16] studies the performance of packets broadcast in a real-world presence-awareness application with PDAs in a large two-hop ad hoc network. Users have no other contact except broadcasting to exchange personal profile. It shows that there is significant intrinsic delay and fluctuation in transmission times even for re-broadcast by a single PDA. It deduces that Probability-based control should allow adequate performance after carefully analysing test data. This application does not consider the mobility of users; users just contact others with wireless interface. In [17] a more complicated platform for future distributed applications is described. A new many-to-many communication protocol (M2MP) is introduced. Though it is a packet broadcasting protocol, packets sent by M2MP are processed based on their content rather than their address.

6 CONCLUDING REMARKS

In this paper we presented protocols to support auction applications in a small ad hoc network. We believe that it is important to develop a variety of interactive applications using ad hoc networks in order to understand what functionality can suitably be incorporated in the mobile computing middleware. In the case of this particular auction application, we described how an application specific ad hoc network can be formed and maintained in an economical manner. Our experiments show that the counter-based flooding algorithm works quite effectively to quench unnecessary broadcast packets in the network. The tests indicate that the system developed has the capability of supporting fair, real-time auctions. We intend to carry out further tests with much larger number of nodes and experiment with adaptive mechanisms for coping with mobility. In our present design, we cope with message loss due to mobility (a source path becoming invalid) simply by relying on frequent broadcasts from the auctioneer. An alternative would be for bidders to maintain redundant paths. In our current design, we have not examined security issues. For example, how to prevent bidders from tampering with messages they are relaying. Such issues are left for future work.

REFERENCES

[1] M. Banatre and F. Weis, "SIS: a new paradigm for mobile communication systems", IST'99, Helsinki, November 1999.

[2] M.P. Wellman and P.R. Wurman, "Real time issues for Internet auctions", IEEE workshop on dependable and real time e-commerce systems (DARE-98), Denver, June 1998.

[3] David B. Johnsson, David A. Maltz, Yih-Chun Hu , "The Dynamic Source Routing Protocol for Mobile Ad hoc networks (DSR)", Internet Draft, draft-ietf-manet-dsr-08.txt, Feb. 2003. Work in progress. http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-08.txt

[4] S.Y. Ni, Y.C. Tseng, Y.S. Chen and J. P. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network", Proceedings of Mobile computing (MobiCom), pp. 151-162, August, 1999.

[5] B.Willams and T.Camp, "Comparison of broadcasting techniques for Mobile ad hoc networks", Proceedings of ACM symposium on Mobile Ad Hoc Networking and Computing (MobiHoc) 2002, pp 194 -205, July 2002.

[6] Charles E. Perkins, Elizabeth M. Belding-Royer, Samir R. Das, "Ad Hoc On Demand Distance Vector (AODV) Routing", Internet Draft, draft-ietf-manet-aodv-13.txt, Feb. 2003. Work in progress. http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-13.txt

[7] Charles E. Perkins and Pravin Bhagwat, "Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers", In Proceedings of the SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications, pages 234–244, August 1994. A revised version of the paper is available from http://www.cs.umd.edu/projects/mcml/papers/Sigcomm94.ps.

[8] Vincent D. Park and M. Scott Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks", in the proceedings of the IEEE INFOCOM 1997, pp 1405–1413, April 1997.

[9] Zygmunt J. Haas, Marc R. Pearlman, Prince Samar, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks", Internet draft, draft-ietf-manet-zone-zrp-04.txt, July, 2002. Work in progress. http://www.ietf.org/internet-drafts/draft-ietf-manet-zone-zrp-04.txt

[10] M. K. Marina and S. R. Das, "Performance of Route Caching Strategies in Dynamic Source Routing," in Proceedings of the 2nd Wireless Networking and Mobile Computing (WNMC), Phoenix, April 2001.

[11] J. Broch, D.Maltz, D.Johnson, Y.-C, Hu, J.Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols", in Proceedings of IEEE/ACM MOBICOM'98, pp 85-97, Oct., 1998.

[12] S.R.Das, C.E. Perkins, E.M. Royer, "Performance comparison of two- on-demand routing protocols for ad hoc networks", in Proceedings of the IEEE INFOCOM 2000, Mar., 2000.

[13] H. Lim, and C Kim, "Multicast tree construction and flooding in wireless ad hoc networks", in Proceedings of the ACM MSWIM 2000.

[14] W. Peng and X.Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks", in Proceedings of MobiHoc 2000.

[15] W.Lou and J.Wu, "On reducing broadcast redundancy in ad hoc wireless networks", in IEEE Transaction on Mobile Computing, pp 111-122, April-June, 2002.

[16] P. Poupyrev, M.Kosuga and P. Davis, "Analysis of wireless Message Broadcast in Large Ad Hoc Networks of PDAs", in the proceedings of The Fourth IEEE Conference on Mobile and Wireless Communications Networks (MWCN 2002).

[17] Alan Kaminsky, "Infrastructure for distributed applications in ad hoc networks of small mobile wireless devices", Technical Report, Rochester Institute of Technology, IT Lab, May 22, 2001

[18] M. Kumar, S.J. Feldman, "Internet Auctions", Proc. 3rd USENIX Workshop on Electronic Commerce, Boston (MA), Aug. 31 - Sept. 3, 1998, pp. 49-59.

[19] Ching-Shan Peng, Jose Miguel Pulido, Kwei-Jay Lin and Douglas M. Blough, "The Design of an Internet-based Real-Time Auction System"

[20] F.Panzieri, S.K. Shrivastava, "On the Provision of Replicated Internet Auction Services", Proc. Of 18th Symp. On Reliable Distributed Systems, Lausanne, 19th October, 1999, pp. 390-395.