Acquaintance Based Soft Location Management (ABSLM) in MANET

Joy Ghosh, Sumesh J. Philip, Chunming Qiao
Department of Computer Science and Engineering
University at Buffalo, The State University of New York
201 Bell Hall, Buffalo, NY 14260-2000
Email:{joyghosh, sumeshjp, qiao}@cse.buffalo.edu

Abstract— **A major challenge faced in Mobile Ad Hoc Networks (MANET) is locating devices for communication, especially with high node mobility and sparse node density. Present solutions provided by ad hoc routing protocols range from flooding the entire network with route requests, to deploying a separate location management scheme to maintain a device location database. In this work, we propose a novel scheme called Acquaintance Based Soft Location Management (ABSLM) in MANET. In ABSLM, nodes make use of the real life concept of making acquaintances and keeping in touch with them regarding each other's current locations. ABSLM has a two-fold aim: to avoid the overhead of flooding; and to use a 'soft' location management setup that does not require strict location management strategies and is thus computationally less expensive than standard 'hard' location management schemes. Simulation results show that ABSLM not only outperforms existing flooding schemes (such as DSR [1] and LAR [2]) in terms of throughput, overhead and location discovery latency, but also achieves performance comparable to 'hard' grid based location management schemes (like SLALoM [3]) with a much lower control overhead.**

I. INTRODUCTION

A wireless mobile ad hoc network (MANET) is an infrastructure less group of wireless nodes that forward packets for one another. The hurdles in designing routing protocols for such networks are node mobility, limited node power, and restricted bandwidth of the wireless medium. Traditional routing protocols developed for wired networks like Distance Vector and Link State routing do not consider these limitations, and hence are unsuitable.

Routing protocols for MANET can be broadly classified into two categories: *proactive* and *reactive*. Proactive protocols maintain an up-to-date image of the network topology at each node by broadcasting the neighborhood information of each node frequently throughout the network. In MANET where the network changes fast, such an effort is costly in terms of bandwidth and power consumption. This also leads to congestion and poor throughput. On the other hand, discovering and maintaining a route to the destination for every new session is a major challenge for reactive protocols. For example, Dynamic Source Routing (DSR) [1] uses pure flooding technique to locate destinations. Location Aided Routing (LAR) [2] tries to fair better than DSR by estimating the approximate region containing the destination (from some prior location information) and flooding only within that expected zone. However all such source routed protocols suffer significantly from path breakage caused by node mobility since there is a high probability of nodes in the path moving out of range of transmission of the preceeding node. Repeated efforts in finding a path between a source and destination pair increases the overhead to maintain a single session.

Recently, a new family of protocols [3], [4], [5], [6] has been proposed for MANET which makes use of *location management* and *geographic forwarding* for routing data packets, where nodes are aware of their own location via the use of a GPS receiver. When a source needs to communicate with a destination, it tries to find the geographical coordinates of the destination using a location service. Once the location is obtained, each packet is forwarded to one of the source's neighbors which is geographically closer to the destination than the source itself. The neighbor that receives the packet forwards it in a similar fashion till the packet finally reaches the destination. Geographic forwarding lends itself as an attractive candidate for routing in MANET, since the amount of network topology information that needs to be stored by each node is minimal and since the data packets need not store the entire route within them anymore. Moreover, since an end-to-end session does not depend on any specific route, the effect of link breakage on a session is greatly reduced.

However, the challenge now lies in effectively maintaining a location database of all nodes in the network. SLALoM [3] is one such location management scheme that operates by partitioning the network terrain into well numbered grids. A hierarchy is then setup to have bigger regions comprising of smaller ones. All nodes are then assigned multiple home regions distributed evenly throughout the network such that a given node's location is known to all nodes in that node's home region. Each node is also responsible for updating its home regions with its current location information. Home regions that are near are updated more frequently than those that are far. Thus a source node just needs to query one of the home regions of the destination (that can be computed by the source locally) which is closest to the source, to get the destination location information.

Deployment of such schemes in MANET requires a fair amount of node density to prevent the occurance of empty home regions. In networks with highly mobile nodes, these schemes will incur considerable overhead in terms of location updates and maintenance of a distributed location database even with low traffic. Thus there is a wide gap between the low maintenance, low performance flooding based protocols and the high maintenance, high performance location management based protocols.

In this work, we propose a novel framework called Acquaintance Based Soft Location Management (ABSLM) that makes use of a simple heuristic to maintain a distributed location database for efficient geographic routing in MANET. A main difference of our scheme from the location management schemes mentioned above is that we make no assumptions regarding either the shape or size of the network terrain or the density of the nodes. By the aid of acquaintances, our scheme informally builds a 'soft' location management setup and eliminates the need for strict management strategies (like computation of grids, assignment of location servers or home regions, location registration, etc). Our main idea is inspired by the experimental studies in social sciences by S. Milgram [7]. In this pioneering work, the author spoke of the likelihood of two individuals in a social network to be connected through a short sequence of intermediate acquaintances. On repeated experiments amongst different pairs of strangers in USA, the average length of the intermediate sequence came out to be around 5 or 6. This later became popular as the principle of 'six degrees of separation' [8]. In their study however, the source was informed about the approximate location of the destination and the source had to forward the packet to acquaintances known to itself only on a first name basis. In our case, the source has no idea regarding the location of the destination, but it uses acquaintances to search for the destination with the hope of finding them being connected via a short list of common contacts, as observed by the studies mentioned above. Additionally, while other protocols (like DSR and LAR) struggle to cope with the challenges imposed by mobility, ABSLM uses the mobility of the nodes to its advantage for achieving better performance. Thus on one hand, ABSLM uses geographic forwarding to avoid the limitations of flooding protocols, and on the other hand due to the 'soft' location services provided, it achieves performance comparable to location management schemes (like SLALoM) with a significantly lower control overhead. To the best of our knowledge, no other solution has been proposed in this capacity that tries to utilize this concept of acquaintanceship amongst nodes, to bring about a soft location management based routing protocol for wireless ad hoc networks.

The rest of the paper is outlined as follows: In Section II, we describe the details of the ABSLM protocol. In Section III, we analyze ABSLM and compare it with DSR, LAR and SLALoM, and present our simulation results. In Section IV we present a detailed survey on other related work and conclude this work in Section V.

II. THE PROPOSED ABSLM PROTOCOL

The basic idea of ABSLM is to make use of the real life concept of making acquaintances and keeping in touch with them regarding each other's current locations to bring about a 'soft' location management system. When a source needs to obtain the location of a destination, it queries its acquaintances about it. If the destination is unknown to the acquaintances, they in turn ask their acquaintances and the process is repeated until the destination's location is discovered. Thus, ABSLM makes use of a location database which is maintained among the nodes in a distributed fashion. In the following subsections, we outline the different phases that constitute the ABSLM protocol.

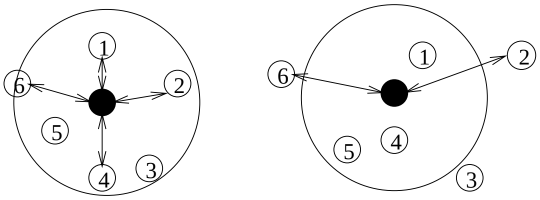
A. Acquaintance Selection

Each node periodically broadcasts its current location so that all nodes are aware of their neighbors and their respective locations. At start, each node randomly selects a neighbor who is not already an acquaintance, and sends an acquaintanceship request (*ACQ_REQ*). Simultaneously, it adds this neighbor in its acquaintance table as a *pending* entry. If the node does not get an acceptance from the *pending* acquaintance after a specific period of time, it deletes the entry from its acquaintance table, and retries the acquaintance selection process with a new neighbor. If the neighbor does accept the invitation (*ACQ_ACC*), the node marks that entry in its acquaintance table as *accepted*. As far as using acquaintances to locate a destination is concerned, a node does not distinguish between its *pending* and *accepted* acquaintances. The node however, sends location updates (*LOC_UPD*) containing its up-to-date location only to its *accepted* acquaintances.

We fix the maximum number of acquaintances that a node can have so that the memory requirements to keep track of acquaintances and the overhead to maintain acquaintanceship and discover destinations are practically manageable. Thus, any node that receives an acquaintanceship request replies with an acceptance only if it can accommodate a new acquaintance. In the event that a node receives an acquaintanceship request from a node to which it had send an acquaintanceship request previously, it marks the corresponding entry in its acquaintance table to *accepted*. In case of rejection, the node just ignores the request and lets the requesting node time out.

B. Acquaintanceship Termination

If a node finds any of its acquaintances to be in its neighborhood for a period of time greater than a timeout interval, it deletes that entry from its acquaintance table. Since both acquaintances will sense each other in their neighborhoods, the acquaintanceship will terminate mutually without the need for explicit termination messages. The timeout interval is chosen to be the time it takes for a node to move in and out of its acquaintance's transmission range at a speed equal to the average velocity of the nodes in the network. This method of initiating and maintaining acquaintanceship is illustrated in Figure 1. The intuition behind this form of acquaintanceship termination is for nodes to have acquaintances spread all over the network which, in turn, leads to faster discovery of a destination (see Section II-E for more detailed description).



(a) Acquaintance initiation (b) Acquaintance after timeout

Fig. 1. (a)Initially make acquaintances with neighbors (b)After a timeout, only those nodes that move out of the neighborhood remain as acquaintances

C. Location Update

Each node checks its own location periodically, and on moving a distance equal to its transmission radius, sends out a location update (*LOC_UPD*) to each of its *accepted* acquaintances. Additionally, the node sends a location update to each node with whom it has an ongoing data connection. On getting location updates from its acquaintances, a node updates its acquaintance table with the location information contained within the updates.

D. Nosy Neighbors

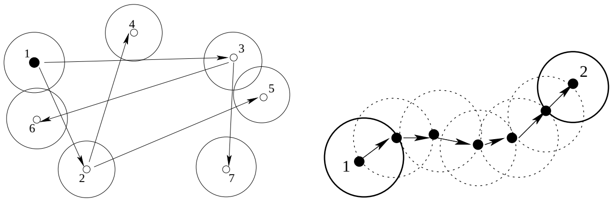
As an optimization attempt, each node also maintains a location cache whose entries consist of the locations of the nodes (other than its acquaintances) this node learned of, while forwarding packets for other nodes in promiscuous mode. However, the information contained by only two types of control packets are relevant:

1) *Location Update*
   While forwarding a *LOC_UPD* packet, each node inserts (or updates) the location information contained within the packet into its cache. Additionally, if the node has room to make a new acquaintance, it adds a new entry for this node in its acquaintance table, and sends an acquaintanceship request to it. Intuitively, the cache allows for faster discovery of destination nodes during the discovery phase. To prevent nodes from using stale cache entries, entries from the cache are deleted after a timeout interval.
2) *Location Response*
   If a node knows about a queried destination, it geographically forwards a response (*LOC_RESP*) back to the source containing the location of the destination. This node may know of the destination for one of the following reasons:
   - the destination is a neighbor or an acquaintance
   - there is a live connection with the destination
   - destination's information was in the location cache

An intermediate node while forwarding this *LOC_RESP* packet looks inside and may cache the location information of both the source and the destination.

E. Acquaintance Based Location Discovery and Data Transfer

At the start of communication, the source checks its neighbors, acquaintances, and location cache in that order for information regarding the destination. If there is no such information, it buffers the data packet and creates a location query (*LOC_QRY*) with an unique sequence number. It also specifies a maximum logical hop count in the query, which specifies the number of times a query packet can be forwarded by acquaintances to other acquaintances during the discovery process. The query is then geographically forwarded to all its *pending* and *accepted* acquaintances. Figure 2 illustrates the case where the maximum logical hop is 2. If the node has no acquaintances, it broadcasts the query to all its neighbors. This type of broadcasting in the absence of acquaintances is done only at the source node to enable a location query to be initiated properly. The neighbors in turn forward the query to all their acquaintances. In the absence of acquaintances, a neighbor (other than the source) will drop the query. Thus, ABSLM avoids full flooding of location query packets.



(a) Querying 2 hops away (b) 1 logical hop

Fig. 2. (a) Node 1 queries 2 and 3 by one logical hop each. Node 2 queries 4 and 5, and node 3 queries 6 and 7 by one logical hop each (b) single logical hop from 1 to 2 by geographic forwarding (the larger circles in both figures, represent the transmission range of each node)

An intermediate node (nosy neighbor), while forwarding query packets from a node to its acquaintance, sniffs into the packet to see if it knows of the destination. If it does, it creates a *LOC_RESP* containing the location of the destination, and geographically forwards the response back to the source (as described earlier). When an acquaintance gets a *LOC_QRY*, if it knows of the destination, a *LOC_RESP* containing the location of the destination is sent back to the source. Otherwise, it decrements the logical hop count by unity, and if it is non-zero, geographically forwards the query to all its acquaintances. However, a node does not forward the query to an acquaintance if:

- that acquaintance forwarded this node this query
- that acquaintance is the source of this query

On receiving a *LOC_RESP*, the source retrieves all the buffered data for that destination and forwards them geographically to the destination. Duplicate location responses are ignored. It also adds the destination to a list of ongoing connections and appends its location information to each data packet that it forwards to the destination. The destination also adds the source to its list of live connections, and from then on, both nodes update each other of their current location periodically for the duration of the communication session. To further optimize this discovery process, a source while waiting for a *LOC_RESP* keeps a check to see if the destination itself entered the neighborhood. In that case all buffered packets are readily forwarded to that destination and all *LOC_RESP* packets for that particular query are ignored.

III. COMPARATIVE STUDY

To evaluate the efficiency of the protocol, we implemented ABSLM in GloMoSim [9] and compared it with three protocols: DSR [1] (as it represents the most rudimentary of all source routed flooding techniques); LAR [2] (since it utilizes approximate destination location information to restrict flooding); and SLALoM [3] (since we found it to be a good representative of the grid based location management protocols). For additional performance comparison between various other existing MANET routing protocols (including the ones chosen in this paper) the readers are referred to: [10], [11], [12], [13], [14].

We used the standard distribution of DSR and LAR Scheme 1 provided with GloMoSim and implemented SLALoM as described in [3]. The simulation parameters were chosen as follows:

- simulation time = 2000s
- terrain size = 2000 m x 1000 m
- number of nodes = 100
- radio range = 350 m
  (a transmission covers 19.25% of the terrain)
- mobility = Random Waypoint Model
  (velocity from 0 m/s to 10 m/s with pause of 15s)
- traffic = Random CBR
  (packets of size 1 K ; 50 packets/connection; inter packet arrival time uniformly distributed between 1s to 3s)

We performed some simulations of ABSLM alone with the above parameters in an effort to see the effect of maximum number of acquaintances and the maximum number of logical hops on the throughput and control overhead. When the maximum number of acquaintances is greater than 3, we saw a dramatic increase in overhead whereas the throughput remains same. Similarly, allowing query packets to be forwarded for more than 2 logical hops increases the overhead unnecessarily without substantially increasing the throughput. So we empirically fixed the maximum number of acquaintances to be 3, and the query hop count to be 2 for the subsequent experiments.

Figure 3 shows the average throughput for each protocol as a function of traffic load. While both LAR and ABSLM achieve high and stable throughput with increasing traffic load, DSR performs poorly. This is mainly because DSR cannot deal with link failures as efficiently as LAR, whereas ABSLM is not affected much by link breaks due to geographic forwarding as discussed earlier. LAR is much more aggressive in searching for new routes when compared to DSR since LAR continually expands the estimated region of the destination and searches for a path to the destination in that region. SLALoM performs the best since it has the advantage of both geographic forwarding and the complete location database that allows the destination location to be quickly ascertained. ABSLM did not fare as well as SLALoM since it only offers a 'soft' location management setup.

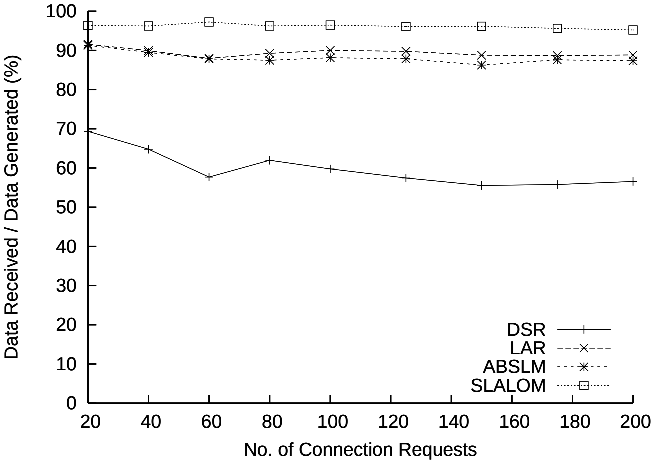


Fig. 3. Data Throughput vs. Load

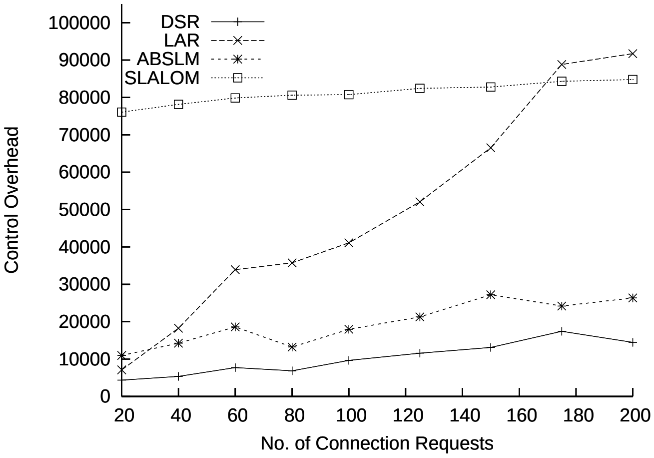


Fig. 4. Control Overhead vs. Load

Figure 4 shows the impact of traffic load on the control overhead in each protocol measured in terms of the number of control packets received at all nodes. Control packets in DSR and LAR consist of 'route requests', 'route responses' and 'route errors'. In ABSLM we consider: periodic *HELLO*, *ACQ_REQ*, *ACQ_ACC*, *LOC_QRY*, *LOC_RESP* and *LOC_UPD*. In SLALoM we have: periodic *HELLO*, *location queries*, *responses*, *updates*, *notifications*, and protocol *maintenance* packets that are needed to inform a node entering a new region, of all the nodes that have that region as their home. In Figure 4 we find DSR to have the lowest overhead since it is purely reactive in that it only needs to broadcast a 'route request' at the start of each session, or if a source route fails. On the other hand, LAR has a very high control overhead which increases with traffic load. This is because in LAR, if a source had heard from a destination previously, it tries to estimate the region where the destination is expected to be found, and floods route requests in that area. Worse, if the request times out, this procedure is carried out repeatedly by increasing the size of the flooding zone which may grow to the entire terrain. This results in many redundant route requests. SLALoM has a high control overhead even for lower traffics just to maintain the location database. From the figure 4, it's clear that the overhead in ABSLM, while much lesser than that of LAR and SLALoM, is a bit more than that of DSR (note that ABSLM has a much better throughput than DSR).

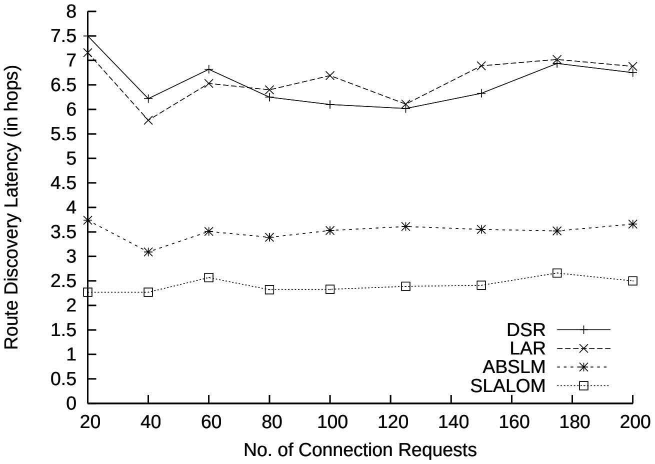


Fig. 5. Route Discovery Latency vs. Load

Figure 5 shows the route discovery latency(in *physical* hops) in each protocol. This metric gives us an estimate of the time a source has to wait before it can initiate data transfer after the request for a session arrives. Since the route request has to reach the destination before a reply can be sent (in the absence of a cached location), route discovery latency is higher in both LAR and DSR on average. In ABSLM, since a *LOC_RESP* can be sent back to the source by an acquaintance (or an intermediate node), node discovery can be much faster, as evident from the figure. SLALoM performs best by virtue of its complete location database, where each node can query a destination's home region for its location. In our simulation scenario almost all queries in SLALoM were serviced successfully by the 'near' home regions, thereby reducing the overall discovery latency considerably.

As a final test to the robustness of each protocol, we also simulated the performance by increasing the mobility of each node. We kept all other simulation parameters same as before but went on to increase only the maximum velocity of nodes in the Random Waypoint model from 10 m/s to 50 m/s and fixed the pause time to 0 s. The traffic was fixed to 200 connection requests. Figure 6 shows the throughput achieved by each protocol with varying mobility. Both LAR and DSR suffer from increased mobility because of increased number of source route failures, which considerably affect their throughput. On the other hand, ABSLM fares much better, since higher mobility enables a node to have acquaintance nodes spread all over the network, thereby ensuring efficient destination discovery. SLALoM is not much affected by mobility either, due to its dependance on geographic forwarding that shields it from link breaks. Overall, ABSLM emerges as a clear winner over LAR and DSR in terms of higher throughput, lower control overhead, lower route discovery latency and performs very well in highly mobile networks. At the same time ABSLM is a viable alternative to extensive location management schemes as it performs nearly as well with a significantly lower control overhead that would lead to lower power consumption and lesser congestion.

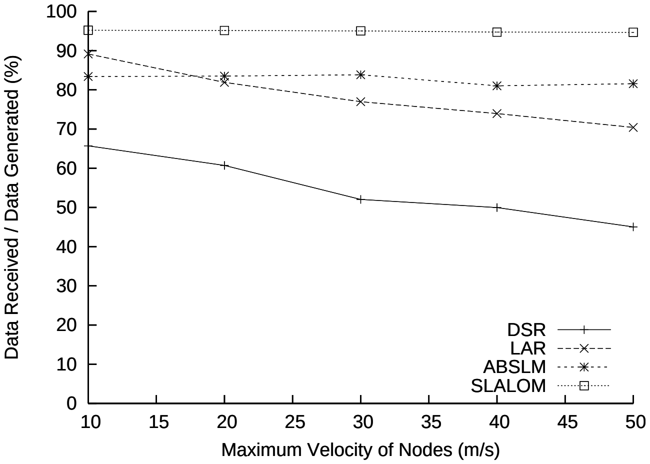


Fig. 6. Data Throughput vs. Mobility

IV. OTHER RELATED WORK

In [15], the authors tried to complement pure reactive approaches by proactive strategies within zones to reduce the location discovery delay and restrict the flooding required. In [16], the reactive part of [15] was supplemented with the GPS based query optimizations. In [17], [18] the advantages of reactive nature of DSR is combined with the location-based solutions to achieve improved efficiency, but they still restrict themselves to source routes and consequently suffer more from link breaks than the protocols based on geographic forwarding. In [19], the authors introduced an efficient way of disseminating geographical position of nodes. By this mechanism, nodes far apart update each other about their own positions less frequently than those closer together, based on the relative velocity. Using this concept several multipoint communication protocols have been proposed, such as [20], [21]. In [22] the authors used a clustering algorithm to restrict the flooding required by DSR. Consequently literature describes several protocols (e.g. [23], [24], [25]) that extends this concept of clustering to propose soultions for multicasting issues in wireless ad hoc networks.

In [26], the authors did extensive analysis of an anchor based routing technique that uses a concept of 'anchors' to loosely source route messages towards the destination. In their work, they tried to make geographic routing more efficient by finding some fixed geographic locations (not nodes) enroute to the destination that are called 'anchors'. Packets are forwarded geographically towards an anchor point until it reaches a node close enough to that point. Thereafter its forwarded to nodes that take the packet towards the next anchor point, till it reaches the destination node. They used special nodes called

Friend Assisted Path Discovery (FAPD) [27] responders that maintain friendship amongst themselves to learn about the irregular network topology and to help sources find appropriate anchor points towards their respective destinations. These responders do not take part in the forwarding path themeselves. Thus, this work differs from ours in the following ways: we do not reserve nodes for specific tasks (like the responders); we make use of acquaintanceship not only to learn collectively about the network neighborhood but to also forward packets to the destination.

We are currently working on a novel approach to selection of acquaintances, based on some activity index that will grade the attractiveness of each node as an acquaintance to others. We are comparing this scheme with the current approach of randomly selecting neighbors as acquaintances. In addition to this, we are trying to optimize the geographic forwarding technique to include power aware concepts in choosing the next hop neighbor. We hope to achieve longer network lifetime through this modification.

V. CONCLUSION

We have described a few popular schemes for routing in mobile ad hoc networks, and show that these schemes perform rather poorly in networks with high traffic volume and node mobility. Geographic routing may be a possible candidate for routing in such networks, and we propose a novel scheme known as Acquaintance Based Soft Location Management (ABSLM), in which nodes use the real life concept of making acquaintances and keeping in touch with them regarding their current locations, for efficient geographic routing. ABSLM, inspired by the ‘small-world phenomenon’ concept, combines the advantages of purely reactive protocols with the concept of location databases to offer a cost effective solution that is intended to bridge the gap between the two extremes of flooding and complete location management schemes. The number of acquaintances that a node can have, and the number of query steps are the two parameters that need to be tuned for the efficient operation of ABSLM. By empirically selecting appropriate values for these parameters, we have shown that ABSLM not only outperforms flooding protocols like DSR and LAR in terms of throughput, discovery latency and control overhead, but also peforms as well as ‘hard’ location management schemes like SLALoM at a significantly lower control overhead.

REFERENCES

[1] J. Broch, D. B. Johnson, and D. A. Maltz, “Dynamic Source Routing in ad hoc wireless networks,” *Mobile Computing, Kluwer Academic Publishers,* pp. 153–181, 1996.

[2] Y. B. Ko and N. H. Vaidya, “Location-Aided Routing (LAR) in mobile Ad-Hoc networks,” *IEEE INFOCOM*, pp. 66–75, October 1998.

[3] C. T. Cheng, H. L. Lemberg, S. J. Philip, E. van den Berg, and T. Zhang, “SLALoM: A Scalable Location Management Scheme for Large Mobile Ad-hoc Networks,” *Proceedings of Wireless Communications and Networking Conference,* March 2002.

[4] J. Li, J. Janotti, D. S. J. D. Couto, D. R. Karger, and R. Morris, “A Scalable Location Service for Geographic Ad Hoc Routing,” *Proceedings ACM/IEEE MobiCom*, pp. 120–130, August 2000.

[5] S.-C. Woo and S. Singh, “Scalable Routing Protocol for Ad Hoc Networks,” *Wireless Networks*, vol. 7, pp. 513–529, January 2001.

[6] S. Philip and C. Qiao, “Hierarchical grid location management for large wireless ad hoc networks,” *ACM SIGMOBILE Mobile Computing and Communications Review (MC2R),* to appear.

[7] S. Milgram, “The small world problem,” *Psychology Today*, vol. 1, p. 61, 1967.

[8] J. Guare, “Six Degrees of Separation,” *A Play*, (Vintage Books, New York) 1990.

[9] X. Zeng, R. Bagrodia, and M. Gerla, “Glomosim: a library for parallel simulation of large-scale wireless networks,” *Proceedings of the 12th Workshop on Parallel and Distributed Simulations – PADS ’98*, May 1998.

[10] S. R. Das, C. E. Perkins, and E. M. Royer, “Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks,” *Proceedings of INFOCOM 2000,* March 2000.

[11] T. Camp, J. Boleng, B. Williams, L. Wilcox, and W. Navidi, “Performance comparison of two location based routing protocols for Ad Hoc networks,” *Proceedings of the IEEE INFOCOM*, 2002.

[12] S. J. Lee, C. K. Toh, and M. Gerla, “Performance evaluation of table-driven and on-demand ad hoc routing protocols,” *Proceedings of IEEE PIMRC ’99*, pp. 297–301, Sept 1999.

[13] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, “Scenario-based performance analysis of routing protocols for mobile ad-hoc networks,” *Proceedings of ACM/IEEE MOBICOM ’99*, pp. 195–206, 1999.

[14] S. J. Philip, J. Ghosh, S. Khedekar, and C. Qiao, “Performance Study of Location Management Protocols for Mobile Ad hoc Networks,” *Submitted to ICC ’04*.

[15] M. Pearlman and Z. Haas, “Determining the Optimal Configuration for the Zone Routing Protocol,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1395–1414, August 1999.

[16] A. Boukerche and S. Rogers, “GPS query optimization in mobile and wireless ad hoc networking,” *Proceedings of Sixth IEEE Symposium on Computer and Communications*, July 2001.

[17] S. Basagni, I. Chlamtac, and V. Syrotiuk, “Dynamic source routing for ad hoc networks using the global positioning system,” *Proceedings of IEEE Wireless Communications and Networking Conference*, vol. 1, pp. 301–305, 1999.

[18] A. Boukerche, V. Sheetal, and M. Choe, “A route discovery optimization scheme using GPS system,” *SCS/ACM/IEEE Annual Simulation Symposium,* April 2002.

[19] S. Basagni, I. Chlamtac, and V. Syrotiuk, “Geographic messaging in wireless ad hoc networks,” *Proceedings of VTC1999*, vol. 3, pp. 1957–1961, Spring 1999.

[20] A. Boukerche, “Simulation based study of on-demand routing protocols for ad hoc wireless networks,” *Proceedings of 34th Annual Simulation Symposium*, pp. 85–92, April 2001.

[21] S. Basagni, I. Chlamtac, and V. Syrotiuk, “Location aware one-to-many communication in mobile multi-hop wireless networks,” *Proceedings of VTC2000*, vol. 1, pp. 288–292, Spring 2000.

[22] M. Jiang, J. Li, and Y. Tay, “Cluster Based Routing Protocol (CBRP) Functional Specification,” vol. work in progress, August 1998.

[23] B. An and S. Papavassiliou, “A mobility-based clustering approach to support mobility management and multicast routing in mobile ad-hoc wireless networks,” *International Journal of Network Management,* vol. 11, pp. 387–395, 2001.

[24] C.-C. Chiang, M. Gerla, and L. Zhang, “Shared Tree Wireless Network Multicast,” *Proceedings of IEEE 6th International Conference on Computer Communications and Networks (ICCCN ’97),* September 1997.

[25] B. An and S. Papavassiliou, “Geomulticast: architectures and protocols for mobile ad hoc wireless networks,” *Journal of Parallel and Distributed Computing*, vol. 63, pp. 182–195, 2003.

[26] L. Blazevic, J.-Y. L. Boudec, and S. Giordano, “A Location Based Routing Method for Irregular Mobile Ad Hoc Networks,” *EPFL-IC Report Number IC/2003/30*, pp. CH–1015, May 2003.

[27] L. Blazevic, S. Giordano, and J.-Y. L. Boudec, “Anchored Path Discovery in Terminode Routing,” *Proceedings of The Second IFIP-TC6 Networking Conference (Networking 2002)*, May 2002.