LANMAR: Landmark Routing for Large Scale Wireless Ad Hoc Networks with Group Mobility

Guangyu Pei, Mario Gerla and Xiaoyan Hong
Computer Science Department
University of California
Los Angeles, CA 90095-1596
{pei,gerla,hxy}@cs.ucla.edu

Abstract* – In this paper, we present a novel routing protocol for wireless ad hoc networks – Landmark Ad Hoc Routing (LANMAR). LANMAR combines the features of Fisheye State Routing (FSR) and Landmark routing. The key novelty is the use of landmarks for each set of nodes which move as a group (e.g., a team of co-workers at a convention or a tank battalion in the battlefield) in order to reduce routing update overhead. Like in FSR, nodes exchange link state only with their neighbors. Routes within Fisheye scope are accurate, while routes to remote groups of nodes are "summarized" by the corresponding landmarks. A packet directed to a remote destination initially aims at the Landmark; as it gets closer to destination it eventually switches to the accurate route provided by Fisheye. Simulation experiments show that LANMAR provides efficient and scalable routing in large, mobile, ad hoc environments in which group mobility applies.

I. INTRODUCTION AND BACKGROUND

As the wireless and embedded computing technology continues to advance, increasing numbers of portable computing and communication devices will be capable of tetherless communications and ad hoc wireless networking. An ad hoc wireless network is a self-organizing and self-configuring network which does not rely on a fixed infrastructure and has the capability of rapid deployment in response to application needs. Ad hoc networks are very attractive for tactical communication in military and law enforcement. They are also expected to play an important role in civilian forums such as convention centers, conferences, and electronic classrooms. Node mobility, potentially very large number of nodes, and limited communication resources (e.g., bandwidth and power) make routing in ad hoc networks extremely challenging. In particular, the routing protocols for ad hoc wireless networks must quickly adapt to frequent and unpredictable topology changes and must be parsimonious of communications and processing resources.

Existing wireless routing schemes can be classified into two categories according to their design philosophy: (a) proactive (e.g., distance vector or link state based); and (b) reactive (e.g., on demand). Proactive schemes compute global routes in the background. Historically, the first routing scheme used in the early packet radio network, PRNET, was a proactive, distance vector type [5]. The distance vector approach is simple but in a mobile scenario it suffers from slow convergence and tendency of creating loops. These problems can be resolved by the Link State (LS) approach, which is widely used in wired nets (e.g., Internet [17] or ATM [1]). In Link State, global network topology information is maintained in all routers by the periodic flooding of link state updates by each node. Any link change triggers an immediate update. As a result, convergence to a new topology is faster and preventing loops is easier due to global topology knowledge. Unfortunately, excessive control overhead is generated by LS dissemination when high mobility triggers frequent updates. Moreover, routing table size in both Distance Vector and Link State grows linearly with the number of nodes and thus does not scale to large network size.

In general, when wireless network size and mobility increase (beyond certain thresholds), current "flat" proactive routing schemes (i.e., distance vector and link state) become all together unfeasible because of line and processing O/H. In some application domains (e.g., automated battlefield) the scalability of a wireless ad hoc network is achieved by designing a hierarchical architecture with physically distinct layers (e.g., point-to-point wireless backbone, UAVs and satellites). In other application (e.g., sensor networks), however, such physical hierarchy is not cost-effective. Thus, it is important to find solutions to the scalability problem of an homogeneous ad hoc network strictly using scalable routing protocols. One way to solve this problem in the routing domain is hierarchical routing. A hierarchical version of link state called HSR (Hierarchical State Routing) [19], [20] has proven to be quite effective in large wireless networks [12]. HSR, however, requires complex bookkeeping of hierarchical addresses in the face of mobility. A much simpler version of Link State with hierarchical "flavor" is Fisheye State Routing (FSR) [21], [12]. FSR uses the "fisheye" technique (first proposed by Kleinrock and Stevens [14] for visual displays) to reduce routing update overhead. In FSR, each node progressively slows down the update rate for destinations as their hop distance increases. Entries corresponding to nodes within a smaller scope are propagated to neighbors with a higher frequency. As a result, a considerable fraction of topology table entries (corresponding to remote destinations) are suppressed in a typical update, thus reducing line overhead. This approach produces accurate distance and path quality information in the immediate neighborhood of a node, with progressively less detail as the distance increases. As a packet approaches its destination, the route becomes more precise. As network size grows and mobility increases, how-

This work was supported in part by NSF under contract ANI-9814675, in part by DARPA under contract DAAB07-97-C-D321 and in part by Intel.

ever, routes become stale quickly and delays to nodes afar tend to grow large. Moreover, routing table size grows linearly with network size.

A recent approach intended to remedy the scalability and routing overhead problems is the reactive, on demand routing. Several on demand schemes have been proposed including AODV [22], DSR [7], TORA [18] and ABR [24] etc. In these "reactive" protocols a node computes a route only when needed. Small Query/Reply packets are used to discover (possible more than one) route to a given destination.

In general, on demand routing exhibits low line and storage O/H even in very large networks as long as mobility is low and traffic is light and is directed to a few destinations. In a very dynamic traffic and mobility pattern however with many different destinations, the repeated route discovery can lead to high overhead. In fact as mobility increases, the precomputed route may break down, requiring repeated route discoveries on the way to destination. Route caching becomes ineffective in high mobility and with increasing number of destinations. Since flooding is used for query dissemination and route maintenance, routing control O/H tends to grow very high [4]. In the case of 100 nodes and 40 sources with uniform traffic pattern, the results in [4] show that both DSR and AODV generate more routing overhead than actual throughput. Similar findings are reported in [12]. Finally, since a route has to be entirely discovered prior to the actual data packet transmission, the initial search latency may impact the performance of interactive applications (e.g., distributed database queries). In multimedia traffic scenarios requiring QoS guarantees, another type of limitation arises. With on demand route discovery, it is impossible to know in advance the quality of the path (e.g., bandwidth, delay, etc) prior to call setup. Such a priori knowledge (which can be easily obtained from proactive schemes) is very desirable in multimedia applications, as it enables effective call acceptance control without probing the network each time.

A recent proposal which combines on demand routing and conventional routing is Zone Routing Protocol (ZRP) [8], [9]. For routing operations inside a local zone, an arbitrary proactive routing scheme (e.g., distance vector) can be applied. For interzone routing, on demand routing is used. The advantage of zone routing is its scalability, as routing table size $=$ zone size and "global" routing table overhead is thus limited. For interzone routing, however the on-demand solution poses the above mentioned problems of connection latency and high routing load for dense traffic patterns.

Recently, the introduction of low cost GPS [13] technology has stimulated the development of "location" assisted on demand routing schemes [15], [16]. The performance is somewhat improved (especially if mobility prediction is employed), but the basic limitations with respect to high loads still remain.

In this paper, we introduce a new "proactive" routing scheme called Landmark Ad Hoc Routing (LANMAR). It is a combined link state and distance vector routing protocol which exploits and adapts to the wireless ad hoc environment. The concept of Landmark routing was first introduced in fixed wide area networks [25]. The original scheme required predefined multi-level hierarchical addressing. Our scheme does not require predefined hierarchical address, but it borrows the concept of Landmark and extends it to the mobile environment. Specifically, it exploits the landmark concept to handle group mobility. LANMAR helps solve both scalability and mobility problems while keeping line and storage overhead (O/H) low. The rest of the paper is organized as follows. In section II, we describe the LANMAR routing scheme. Section III presents the performance results and section IV concludes the paper.

II. LANDMARK AD HOC ROUTING

A. Network Data Structures

Each node has a unique "physical" identifier (e.g., a 48 bit ethernet address) and landmark ON/OFF flag. In addition, each node has a unique "logical" identifier defined by a subnet field and a host field. The host field is unique in the subnet and might in fact coincide with the physical address. For each node i, one list and three tables are maintained. They are: a neighbor list A_i, a topology table TT_i, a next hop table $NEXT_i$ and a distance table D_i. Each destination j within fisheye scope has an entry in table TT_i which contains two parts: $TT_i.LS(j)$ and $TT_i.SEQ(j)$. $TT_i.LS(j)$ denotes the link state information reported by node j. $TT_i.SEQ(j)$ is a time stamp indicating the "age" of the link state information. Similarly, for every destination j which is within its fisheye scope or which is a landmark node, $NEXT_i(j)$ denotes the next hop to forward packets destined to j on the shortest path. $D_i(j)$ denotes the distance of the shortest path from i to j. Additionally, one or more link weight functions may be defined and used to compute the shortest path based on a specific metric, possibly with constraints. For instance, a bandwidth function can be used to support QoS routing. In this paper, we limit ourselves to min hop paths, thus the link weight is 1.

B. The Landmark Ad hoc Routing Protocol (LANMAR)

In the original landmark scheme for wired networks proposed in [25], the predefined hierarchical address of each node reflects its position within the hierarchy and helps find a route to it. Each node knows the routes to all the nodes within it hierarchical partition. Moreover, each node knows the routes to various "landmarks" at different hierarchical levels. Packet forwarding is consistent with the landmark hierarchy and the path is gradually refined from top level hierarchy to lower levels as a packet approaches destination.

LANMAR borrows from [25] the notion of landmarks to keep track of logical subnets. A subnet consists of members which have a commonality of interests and are likely to move as a “group” (e.g., brigade in the battlefield, colleagues in the same organization, or a group of students from same class). A “landmark” node is elected in each subnet. The routing scheme itself is modified version of FSR. The main difference is that the FSR routing table contains “all” nodes in the network, while the LANMAR routing table includes only the nodes within the scope and the landmark nodes. This feature greatly improves scalability by reducing routing table size and update traffic O/H.

When a node needs to relay a packet, if the destination is within its neighbor scope, the address is found in the routing table and the packet is forwarded directly. Otherwise, the logical subnet field of the destination is searched and the packet is routed towards the landmark for that logical subnet. The packet however does not need to pass through the landmark. Rather, once the packet gets within the scope of the destination, it is routed to it directly.

The routing update exchange in LANMAR routing is similar to FSR. Each node periodically exchanges topology information with its immediate neighbors. In each update, the node sends entries within its fisheye scope. It also piggy-backs a distance vector with size equal to the number of logical subnets and thus landmark nodes. Through this exchange process, the table entries with larger sequence numbers replace the ones with smaller sequence numbers.

C. Drifters and Isolated Nodes

Typically, all members in a logical subnet are within fisheye the scope of the landmark, thus the landmark has a route to all members. It may happen, however, that some of the members “drift off” outside the scope; for example, a tank in a battalion may become stranded or lost. To keep track of such “drifters”, i.e., to make the route to them known to the landmark, the following modification to the routing table exchange is necessary. Each node, say i, on the shortest path between a landmark L and a drifter l associated with such landmark keeps a distance vector entry to l. Note that if l is within fisheye scope of i, this entry is already included in the FSR table of node i. When i transmits its distance vector to neighbor j, say, then j will retain the entry for member l only if $d(j,l) <$ scope or $d(j,L) < d(i,L)$. The latter condition occurs if j is on the shortest path from i (and therefore from l) to L. This way, a path is maintained from the landmark to each one of its members, including drifters.

The routing storage and processing overhead introduced by the distance vector extension to handle drifters is typically small if the fraction of drifting nodes is small. Consider a network with N nodes and L landmarks, and assume that a fraction F of the members of each logical subnet have drifted. In the worst case, the path length from landmark to drifter is $\sqrt{N}$ (assuming a grid topology). Thus, $\sqrt{N}$ is the bound on the number of extra routing entries required at the nodes along the path to the drifter. The total number of extra routing entries is $\sqrt{N}L(FN/L)$ where N/L is the average logical group size. Thus, the extra storage per node is $F\sqrt{N}$. Now, let us assume that the number of nodes in the fisheye scope = # of landmarks = logical group size = $\sqrt{N}$. Then, the basic routing table overhead per node (excluding drifters) is $3\sqrt{N}$. Thus, the extra overhead caused by drifters is $F/3$. If 20% of the nodes in a group are outside of the landmark scope, i.e., have drifted, the extra routing O/H required to keep track of them is only 7%.

It may happen that some nodes are “isolated”, i.e., group size is 1. If the number of isolated nodes is small, say less than the number of landmarks, they can be handled as landmark nodes. If the number represents a substantial fraction of the total node population, various schemes can be considered. The simplest solution is to run the conventional FSR jointly with LANMAR. Namely, “all” nodes participate in FSR. In addition, the LANMAR distance vector scheme is overlaid on top of FSR. The drawback of this hybrid solution is that FSR routing tables are back to “full network” size, thus, do not scale. A slightly better solution is to treat the isolated nodes outside the scope as landmarks, but with progressively lower routing vector update rate as their distance increases. The isolated node is recognized by the fact that it has a “no followers” flag. This latter scheme is quite appealing since it provides a gradual transition from full landmark performance to full FSR performance (when there are no groups). Note that this version of FSR is a hybrid of Link State (within scope) and Distance Vector (outside of scope). The reader can verify that this hybrid version works correctly.

One drawback of the above “isolated node” solutions is the fact that they do not alleviate the routing table O/H when a large fraction of nodes is isolated. There are different ways to handle this problem. One way is to compute routes to isolated nodes “on demand” using existing on-demand routing schemes. This scheme reminds us of ZRP, although in our case we can handle logical groups efficiently. Another approach (which is inspired to Mobile IP) is to maintain a proper number of Home Agents (or “designated” landmarks) such that each isolated node registers with the associated Home Agent and continuously reports to it such that a route is maintained from Home Agent to isolated node. The mechanism for maintaining a route from Home Agent to each of the isolated nodes is the same as the one for Landmarks and drifters. Association to Home Agent can be determined uniquely by hashing the isolated node logical address, for example. Using the previously mentioned grid topology model, and assuming all nodes are isolated and $\sqrt{N}$ Home Agents are elected, one finds that each node requires $\sqrt{N}$ routing entries on the path to its Home Agent (thus, a total of $N\sqrt{N}$ entries in the entire network). Moreover, each node must maintain a route to each of the Home Agents (thus, $N\sqrt{N}$ entries in the entire network). The total number of routing entries per node is thus $O(\sqrt{N})$.

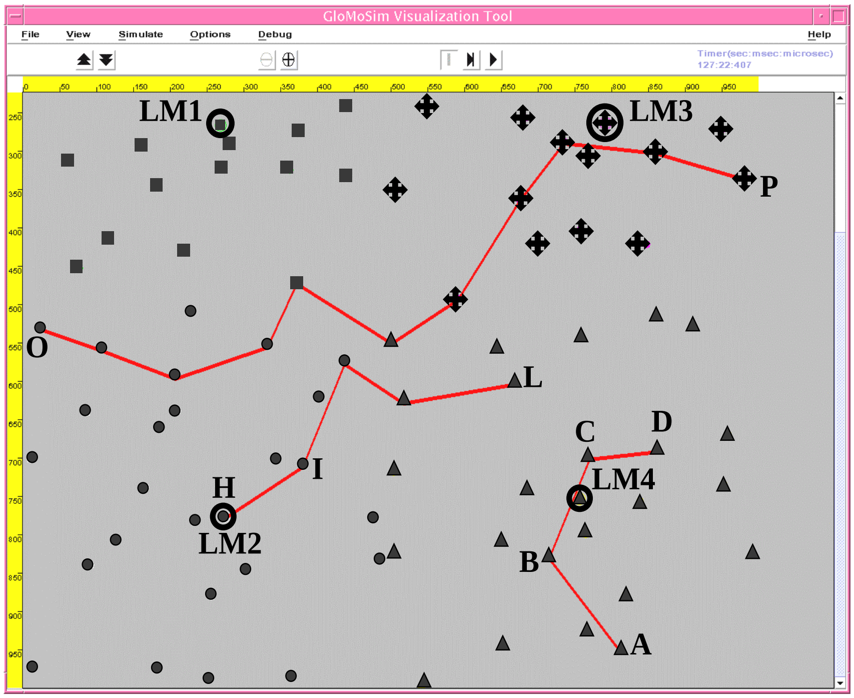

Fig. 1. Snap shot of LANMAR

This compares favorablely with the basic storage O/H of LANMAR, which is $3\sqrt{N}$. Naturally, the route "indirection" increases (almost doubles) path length and thus delay. It also increases overall network traffic.

All of these "isolated node" schemes have their pros and cons. Their evaluation is now in progress. An extensive discussion of the isolated node support is clearly beyond the scope of this paper. It is important to point out, however, that the LANMAR scheme is robust in the face of changes in isolated/group node mix. With proper extensions LANMAR performs quite well (in fact, no worse than conventional FSR) even when all nodes are "isolated".

D. An Illustration of LANMAR

Fig. 1 is a snapshot of a LANMAR simulation run. In this example, there are four logical groups and each group has its landmark node LM_i. The fisheye scope radius is 2. We show three different paths. The first path is from node A to D. D is within the fisheye scope of node B (2 hops from B); thus, B will include D in the routing update sent to A. A has complete routing information about D and can deliver the packets along the shortest path to D. The second path is from H to L. H does not know the route to L. It routes the packets towards the landmark of node L (LM_4) through I. Node I does have a route to L, and can forward the packets directly on the shortest path. A third, much longer path is also shown (from O to P). The path leads first to the landmark (LM_3) of node P. As the packet approaches LM_3, it obtains the direct route to P. It thus bypasses LM_3 and reaches P in two hops. The visual inspection of the LANMAR generated paths shows that they are in all cases quite close (if not identical) to shortest paths.

III. PERFORMANCE EVALUATION

A. Simulation Model

The simulator for evaluating routing protocols was implemented within the GloMoSim library [23]. The GloMoSim library is a scalable simulation environment for wireless network systems. It is based on the parallel, discrete-event simulation language PARSEC [2]. In our experiments, the MAC layer is implemented using the distributed coordination function (DCF) of IEEE 802.11 [11]. More precisely, Request-To-Send (RTS) and Clear-To-Send (CTS) control packets are used to provide virtual carrier sensing for *unicast* data packets to overcome the well-known hidden terminal problem. Each data transmission is followed by an ACK. *Broadcast* data packets are sent using CSMA/CA mode. The radio model is consistent with existing commercial offerings (e.g., Lucent's WaveLAN). Radio transmission range is 150 meters and channel capacity is 2 Mbits/sec. The simulation area is 1000 × 1000 meter square. Each simulation executed for 20 minutes of simulation time.

B. Traffic Pattern and Mobility Models

The source-destination pairs are randomly selected over the network. Their number is varied in the experiments to change the offered load. Traffic is UDP. The interarrival time of the data packets on each source/destination connection is 2.5 seconds. This is a reasonable model of an interactive environment. The size of the data payload is 512 bytes. The load in the network is increased by increasing the number of connections (each with fixed traffic rate), instead of keeping the number of connections constant and increasing the rate on each connection as previously done in [4]. This is a departure from previous simulation studies [4], [3], [6] which focused on performance evaluation for small number of traffic pairs (up to 40 pairs) each with relatively high data rate (3 - 4 pkts/sec).

The mobility model is the *Reference Point Group Mobility* model [10] applied to a square field. Each node in a group has two components in its mobility vector: the individual component and the group component. The individual component is based on the *random waypoint* model [7], [3]. A node randomly picks a destination within the group scope and moves towards that destination at a fixed speed. Once the node reaches the destination, it selects another destination randomly and moves towards it after a 10-second pause time. This behavior is repeated for the duration of the simulation. The speed varies between 2 and 10 m/sec. Pause time is not considered in computing node speed. The group component of mobility is also based on the random waypoint model. In this case, however, the destination is an arbitrary node in the entire system. We use a relatively short pause time of 10 seconds to force the topology change quite dynamically and thus challenge the routing algorithms.

C. Performance Metrics

We have used the same metrics as proposed in [4] to compare protocol performance, namely: (i) *Packet delivery fraction* – the ratio between the number of received data packets and those originated by the sources. (ii) *Average end-to-end packet delay* – the time from when the source generates the data packet to when the destination receives it. The delay includes: route acquisition latency, processing delays at various layers of each node, queuing at the interface queue, retransmission delays at the MAC, propagation and transfer times. (iii) *Normalized routing load* – the number of routing control packets transmitted per data packet delivered at the destination. Each hop-wise transmission of a routing control packet is counted as one transmission. (iv) *Throughput* – the actual throughput achieved to destination.

The first two metrics reflect the routing effectiveness of a protocol and are the most important metrics for best effort traffic. The routing load metric shows the routing control penalty involved in delivering data. These metrics are not independent and should be interpreted jointly. Consider, for example, average end-to-end delay and packet delivery fraction. Packets traveling to destinations many hops away are the most likely to be dropped; the dropped packets are not included in the delay computation. Thus, low delay does not necessarily mean that the route selection is effective: if the delivery fraction is low (i.e., if the drop rate is high), there is a good chance we are measuring delay on short paths only.

D. Simulation Results

In our experiments, we compare the performance of LANMAR with that of FSR, in order to determine the improvement introduced by Landmark routing and its ability to track group mobility. As a reference, we also report the performance of two popular on demand routing schemes implemented in the GlomoSim library, i.e., AODV and DSR. The network size is 100 nodes. The number of logical groups is 4 for LANMAR. The Fisheye scope for both FSR and LANMAR is 2 hops and the update frequency within the fisheye scope is the same for both schemes, i.e., one update every 5 seconds.

The first experiment (Fig. 2) reports the packet delivery fraction under various traffic loads with different speeds. Comparing FSR and LANMAR routing, we note that at zero speed the performance is approximately identical. As speed increases, LANMAR performs remarkably better than FSR. This is mainly due to the fact that the routes to landmarks in LANMAR are updated with the same frequency as the routes within the fisheye scope in FSR. This is much higher than the frequency of route updates to remote nodes in FSR. This indicates that it is more important to keep accurate routes to **landmark nodes** rather than “blurred” routes to **all nodes**. As the load increases from 10 to 300 pairs, the delivery fraction performance of both LANMAR and FSR remains unchanged. This is an indication that the network is not overload, yet. Namely, the packet loss is due to mobility induced route failures rather than congestion and buffer overflow. As for the on demand routing protocols, under light load (i.e., 10 pairs), AODV outperforms all the other schemes. This is because AODV has more accurate routing information. However, when the traffic load increases, AODV gradually loses its advantage. This is because of the increasing routing O/H under high traffic load, especially due to RREQ packets. Both on demand schemes (i.e., DSR and AODV) are affected by the increase in number of pairs: at zero speed, the delivery fraction drops to $\frac{1}{2}$ for both as the number of pairs increases from 10 to 300. As mobility increases, the delivery degradation is stronger in AODV than DSR. Overall, we found that LANMAR outperforms all the other protocol beyond 30 pairs.

Fig. 3 shows the average packet delay as a function of offered load and mobility. The offered load is increased by increasing the number of pairs. A 800 kbps load corresponds to 500 pairs. As the offered load increases, delay increases because of queue buildup. Here again we note that LANMAR

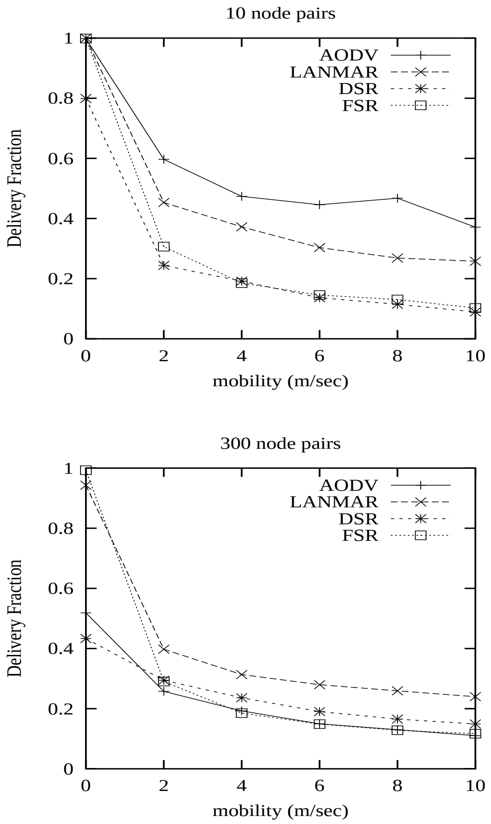

Fig. 2. Delivery Fraction

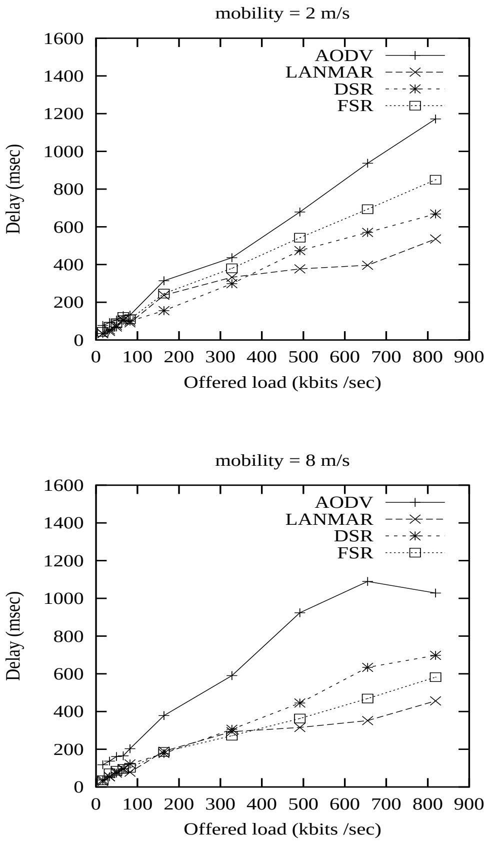

Fig. 3. Delay

yields lower delay than FSR. The accuracy of the route to the landmark proves to be very cost effective, in spite of a possible minor detour (and path increase) to reach the final destination. As for the other schemes, the delay of AODV increases faster than all other protocols because of the higher routing O/H and thus higher load. The delay decrease (across all schemes) when mobility increases is simply due to the fact that packets to distant nodes (i.e., packets which most contribute to delay) are more likely to be dropped.

Fig. 4 reports the normalized routing load. Low routing load is a desirable property for scalability. Recall that the normalized routing load is the ratio of control packets over delivered data packets. For LANMAR and FSR, the number of control packets depends only on topology. It is independent of mobility and number of source/destination pairs. Thus, it is a constant divided by the delivered ratio times the total number of packets generated at sources. Accordingly, the curves for LANMAR and FSR are directly related to the delivery fraction curves in Fig. 2. It is interesting to note that in LANMAR the relative routing O/H is lower than in FSR. In AODV and DSR the number of control packets increases with number of pairs as well as with mobility. For small number of pairs (say 10), the normalized load is comparable across all schemes. As number of pairs and mobility increase, the normalized load of on demand schemes grows much higher. Moreover, AODV has a much higher O/H than DSR. This explains the rapid degradation of AODV in delivery fraction and delay experiments.

Experiments in Fig. 5 demonstrate the effect of loading and mobility on overall throughput. The results are obviously related to those shown in Fig. 2 for delivery fraction. In most cases, the throughput curves grow monotonically with load, thus, the network is not yet saturated. An exception is AODV, which shows early symptoms of saturation at 200 kbps load. The higher the mobility, the lower the throughput. As expected, LANMAR outperforms FSR. DSR outperforms AODV for loads higher than 300 kbps. AODV still performs best under low traffic loads. LANMAR routing on the other hand outperforms all protocols under more stressful situations (higher load and mobility).

Another important performance measure for proactive, routing table based schemes such as FSR and LANMAR is the route storage O/H. Route storage is the critical limitation to scalability of conventional routing schemes (i.e., Distance Vector and Link State). LANMAR and FSR storage requirements can be easily computed analytically. Consider our 100 nodes example. The storage O/H per node is 2600 bytes for FSR and 690 bytes for LANMAR. More generally, consider an ad hoc network with N nodes and $\sqrt{N}$ logical subnets, with nodes

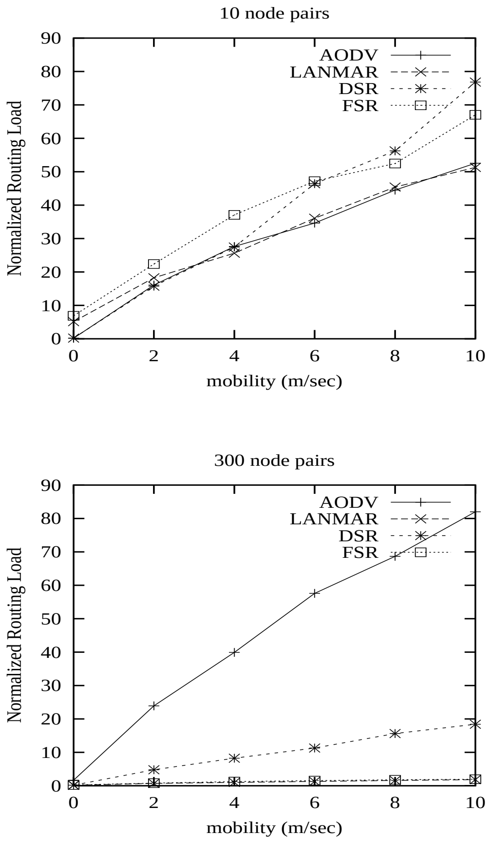

Fig. 4. Normalized Routing Load

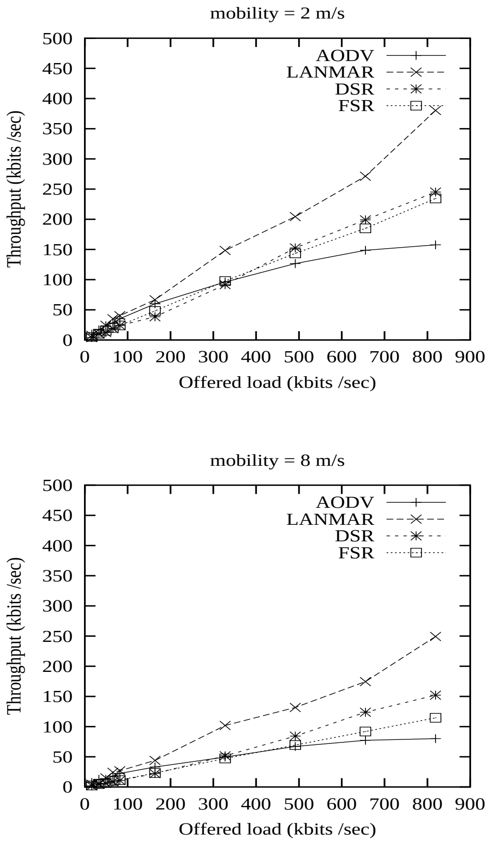

Fig. 5. Throughput vs Mobility

within the "scope" $= \sqrt{N}$. Under these conditions, the FSR storage overhead per node is $O(N)$. Each node must keep a "sparse" connectivity matrix representing the entire network topology ($O(N)$); and a routing vector with all the destinations ($O(N)$). In LANMAR, under the same conditions, each node has a sparse topology matrix and distance vector only for the "scope" nodes ($O(\sqrt{N})$); and an additional routing vector for all the landmark destinations ($O(\sqrt{N})$). Thus, LANMAR routing storage requirement is $O(\sqrt{N})$. Note also that lower storage O/H is coupled with lower route processing O/H and thus lower power consumption, an important consideration in power limited nodes. In summary, when group mobility can be exploited using landmarks, LANMAR performs better than FSR under all measures. It also performs better than on demand routing algorithms, in application scenarios where the traffic pattern is well distributed and load is relatively high.

IV. CONCLUSIONS

In this paper, we have proposed a new routing scheme, Landmark Ad hoc Routing (LANMAR). LANMAR is an extension of Fisheye Routing which exploits group mobility by "summarizing" the routes to the group members with a single route to a landmark. Simulation results show that LANMAR provides an efficient, scalable solution for wireless, mobile ad hoc networks. We have compared performance of our routing protocol with FSR, DSR and AODV. Simulation shows that LANMAR outperforms FSR under all delay and throughput measures. Moreover, LANMAR provides a dramatic reduction in route table storage overhead with respect to FSR. As for the comparison with the popular on demand schemes, we have shown that when the number of communication pairs increases, AODV and DSR will generate considerable routing overhead. Because of this increase in routing O/H, the performance of both AODV and DSR is worse than LANMAR for medium to high traffic loads.

Apart from being effective in group mobility scenarios, LANMAR is robust to shifts in mobility pattern. Even if all nodes move independently (i.e., isolated mobility pattern) LANMAR performs no worse than FSR. Moreover, if the notion of "designated" landmark (i.e., Home Agent) is introduced, the routing storage requirement is much more scalable than in FSR, at a cost of delay increase and throughput reduction. Work is now in progress towards refining many features of the basic LANMAR scheme proposed here. We are working on an algorithm for automatic election of the Landmark nodes. We are also evaluating various alternatives for handling the situations when there is a large fraction of isolated nodes.

References

[1] The ATM Forum, “Private Network-Network Interface Specification v1.0,” 1996.

[2] R. Bagrodia, R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, and H.Y. Song, “PARSEC: A Parallel Simulation Environment for Complex Systems,” *IEEE Computer*, vol. 31, no. 10, Oct. 1998, pp.77-85.

[3] J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, and J. Jetcheva, “A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols,” In *Proceedings of ACM/IEEE MOBICOM’98*, Dallas, TX, Oct. 1998, pp. 85-97.

[4] S.R. Das, C.E. Perkins and E. M. Royer, “Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks”, In *Proceedings of IEEE INFOCOM 2000*, Tel Aviv, Israel, Mar. 2000, pp. 3-12.

[5] J. Jubin and J.D. Tornow, “The DARPA Packet Radio Network Protocols,” Proceedings of the IEEE, vol. 75, no. 1, Jan. 1987, pp. 21-32.

[6] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek and M. Degermark, “Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks,” In *Proceedings of ACM/IEEE MOBICOM’99*, Aug. 1999, pp. 195-206.

[7] D.B. Johnson and D.A. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks,” In *Mobile Computing*, edited by T. Imielinski and H. Korth, Chapter 5, Kluwer Publishing Company, 1996, pp. 153-181.

[8] Z.J. Haas, “A New Routing Protocol for the Reconfigurable Wireless Networks,” In *Proceedings of IEEE ICUPC’97*, San Diego, CA, Oct. 1997, pp. 562-566.

[9] Z.J. Haas and M. R. Pearlman “Determining the Optimal Configuration for the Zone Routing Protocol,” In *IEEE Journal on Selected Areas in Communications*, Aug. 1999, pp. 1395-1414.

[10] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang “A Group Mobility Model for Ad Hoc Wireless Networks,”In *Proceedings of ACM/IEEE MSWiM’99*, Seattle, WA, Aug. 1999, pp.53-60.

[11] IEEE Computer Society LAN MAN Standards Committee, *Wireless LAN Medium Access Protocol (MAC) and Physical Layer (PHY) Specification*, IEEE Std 802.11-1997. The Institute of Electrical and Electronics Engineers, New York, NY, 1997.

[12] A. Iwata, C.-C. Chiang, G. Pei, M. Gerla, and T.-W. Chen, “Scalable Routing Strategies for Ad-hoc Wireless Networks,” In *IEEE Journal on Selected Areas in Communications*, Aug. 1999, pp. 1369-1379.

[13] E.D. Kaplan (Editor), *Understanding the GPS: Principles and Applications*, Artech House, Boston, MA, Feb. 1996.

[14] L. Kleinrock and K. Stevens, “Fisheye: A Lenslike Computer Display Transformation,” Technical report, UCLA, Computer Science Department, 1971.

[15] Y.-B. Ko and N.H. Vaidya, “Location-Aided Routing (LAR) in Mobile Ad Hoc Networks,” In *Proceedings of ACM/IEEE MOBICOM’98*, Dallas, TX, Oct. 1998, pp. 66-75.

[16] S.-J. Lee, W. Su, and M. Gerla, “Ad hoc Wireless Multicast with Mobility Prediction,” In *Proceedings of IEEE ICCCN’99*, Boston, MA, Oct. 1999, pp. 4-9.

[17] J. Moy, “OSPF Version 2,” In *IETF RFC 1583*, 1994.

[18] V.D. Park and M.S. Corson, “A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks,” In *Proceedings of IEEE INFOCOM’97*, Kobe, Japan, Apr. 1997, pp. 1405-1413.

[19] G. Pei, M. Gerla, X. Hong, and C.-C. Chiang, “A Wireless Hierarchical Routing Protocol with Group Mobility,” In *IEEE WCNC’99*,New Orleans, LA, Sep. 1999, pp.1536-1540.

[20] G. Pei and M. Gerla, “Mobility Management in Hierarchical Multi-hop Mobile Wireless Networks,” In *Proceedings of IEEE ICCCN’99*, Boston, MA, Oct. 1999, pp. 324-329.

[21] G. Pei, M. Gerla and T.-W. Chen, “Fisheye State Routing in Mobile Ad Hoc Networks,” In *Proceedings of the 2000 ICDCS Workshops*, Taipei, Taiwan, Apr. 2000, pp. D71-D78.

[22] C.E. Perkins and E.M. Royer, “Ad-Hoc On-Demand Distance Vector Routing,” In *Proceedings of IEEE WMCSA’99*, New Orleans, LA, Feb. 1999, pp. 90-100.

[23] M. Takai, L. Bajaj, R, Ahuja, R. Bagrodia and M. Gerla, “GloMoSim: A Scalable Network Simulation Environment,” Technical report 990027, UCLA, Computer Science Department, 1999.

[24] C.-K. Toh, “Associativity-Based Routing For Ad Hoc Mobile Networks,” *Wireless Personal Communications Journal*, Special Issue on Mobile Networking and Computing Systems, Kluwer Academic Publishers, vol. 4, no. 2, Mar. 1997, pp. 103-139.

[25] P. F. Tsuchiya, “The Landmark Hierarchy: a new hierarchy for routing in very large networks,” In *Computer Communication Review*, vol.18, no.4, Aug. 1988, pp. 35-42.