Abstract—Recently, a new family of protocols has been introduced for large scale ad hoc networks that make use of the approximate location of nodes in the network for geography based routing. Location management plays an important role in such protocols, and previous work in this area has shown that the asymptotic overhead of location management is heavily dependent on the service primitives (location registration, maintenance and discovery) supported by a location management protocol. Currently, SLALoM [1], which is a grid based protocol optimized for large node movements, achieves the best known upper bound on the asymptotic worst case overhead of location management. However, the location registration cost in SLALoM dominates other costs for all practical purposes, and thus novel schemes need to be designed to limit this control traffic. In this work, we use the idea of location forwarding to devise a new scheme called ELF that limits the signalling traffic, and thus enhances the scalability of location management in large ad hoc networks. We find that, while the asymptotic overhead cost by such an improvisation matches that of SLALoM, ELF outperforms SLALoM in average case scenarios.

I. INTRODUCTION

A mobile ad hoc network is a network without preexistent infrastructure. Each node takes on the role of a router for packets not destined for itself. Due to mobility, the network topology varies frequently, and one of the major challenges today is to design a routing scheme that scales well with increasing network size. Traditional routing protocols proposed for mobile ad hoc networks can be classified into categories: proactive [2], [3] and on demand [4], [5], [6]. While they operate well in small networks, these protocols incur heavy signalling traffic for discovery and maintenance of end-to-end routes, which forms a major bottleneck for large networks having node membership in the order of thousands over a large geographic area.

Recently, a new family of protocols [1], [7], [8] has been suggested for large scale ad hoc networks which make use of the approximate location of nodes and use geographic forwarding [7], [9] for routing data packets, where nodes are aware of their location via the use of a GPS receiver. Geographic forwarding lends itself as an attractive candidate for routing in large scale networks, since it does not use pre-computed routes for packet forwarding during a communication session between two nodes. Hence the amount of state information that needs to be stored by nodes is minimal, and link breakage in a route usually does not affect the end-to-end session. However, the source requires the destination node’s location information before it can carry out geographic forwarding. Hence, location management forms an essential entity in protocols that use geographic forwarding, in which nodes periodically update location servers of their current location which can then be queried by source nodes in an on demand fashion for locating destination nodes. Location management usually consists of three phases: location registration (choosing location servers and updating them with location information periodically), location maintenance (periodic cleansing of location database for database consistency) and location discovery (querying location servers to obtain a destination’s location information), and the cost of location management algorithms basically depend upon the cost of the above three primitives.

Under worst case analysis, SLALoM [1] was found to have the best upper bound on the overall asymptotic location management cost among all existing location management schemes. The analysis assumed that each of the location management traffic contributed equally to the overall control traffic (and hence the cost). However, this may not be true, and each of the primitives can affect the flow of control traffic (and thereby the data traffic) in the network differently. For instance, the location registration phase can affect the entire network, since the location update traffic, periodic or triggered, usually traverses the entire diameter of the network, and tends to dominate the overall control traffic in the network. On the other hand, location maintenance is a localized process, and affects only the neighboring nodes of a server node carrying out the maintenance process. In addition, the location discovery phase can be designed not to create a location discovery packet for each data packet that the network layer accepts from the application, by using a location cache which is updated when a location response arrives for the first data packet of a new session. The end points of the live connection can then periodically update each other of location changes to maintain the accuracy of geographic routing. Thus the location discovery traffic can be made proportional to only the number of new sessions in the network. From these observations, we notice that one of the limiting factors in achieving scalability
in location management is the negative influence of the large volume of location update traffic on network resources (power, bandwidth, memory etc.). Hence, novel methods need to be developed to reduce the number of update messages spewed by the location management scheme.

The concept of forwarding has been suggested for reducing network signalling and database loads in PCS networks [10]. The idea here is to set up a forwarding pointer between two location databases when a user moves across a database boundary. An incoming call to the user traverses the chain of forwarding pointers to locate the user. In order to limit the delay for locating a user, a resetting operation is performed on the forwarding chain after $k$ forwarding steps. While this scheme appears intuitively feasible for an infrastructure based system with a dedicated control channel that manages a distributed set of location databases independent of users, we are interested in finding out if (i) a framework for location forwarding can be established for ad hoc networks, (ii) if such a scheme fairs well in a system where location databases are to be managed by nodes themselves, and where the bandwidth is shared between control and data and (iii) if the scheme will help us to reduce the location update overhead to a practically manageable level.

The rest of this paper is organized as follows: Section II briefly describes SLALoM, and then presents a new algorithm which we call Efficient Location Forwarding (ELF) that incorporates our modifications to SLALoM to improve the location update overhead. In Section III, we analyze the asymptotic cost of the proposed protocol. Section IV describes the simulation scenarios and results on the average case performance of both protocols, and conclude this work in Section V.

II. SCALABLE LOCATION MANAGEMENT ALGORITHMS

A. Description of SLALoM

Given a square region of area $A$, SLALoM divides the topography into $G$ logical unit regions (referred to as Order-1 regions), where each node is aware of the size of the topography as well as the size of an Order-1 region. It then combines $K^2$ Order-1 regions to form Order-2 regions. Each node selects a home region in each Order-2 region via a function $F$ that maps roughly the same number of nodes to each Order-1 region in an Order-2 region. Hence, every node has $O(\frac{1}{K^2})$ home regions in $A$ (note that since the original square cannot be perfectly tiled with Order-2 regions, it is possible that some nodes may not have home regions in the Order-2 regions adjacent to the boundary of $A$). Nonetheless, for any nodes $u$ and $v$, it is true that there is a home region of $u$ at most $\sqrt{2K}$ from where $v$ is located. Also, if a node $u$ is present in an Order-1 region $R_i$, which lies in an Order-2 region $Q_i$, then all home regions of $u$ that lie in or adjacent to $Q_i$ are considered near home regions, while the rest are considered far home regions.

Figure 1 shows a sample square topography divided into Order-1 and Order-2 grids, where an Order-2 grid consists of 16 Order-1 grids. The shaded grids indicate node $u$’s home regions in each Order-2 grid. The shaded grid in region $R$ and the eight shaded grids around $R$ represent $u$’s near home regions while the remaining are far home regions.

Location Management in SLALoM: All nodes present in a home region of $u$ act as location servers for $u$, and keep an entry for the location of $u$ in their location database. When $u$ moves across two Order-1 regions $R_i$ and $R_{i'}$, it does the following:

1. If $R_i$ and $R_{i'}$ are in the same Order-2 region $Q_i$, $u$ informs all its near home regions of the movement, by a partial location update.
2. If $R_i$ is in $Q_i$, and $R_{i'}$ is in a different Order-2 region $Q_{i'}$, $u$ updates all home regions of the movement by a full location update.
3. $u$ also requests nodes in $R_{i'}$ about the location information it has to keep for nodes that had selected $R_{i'}$ as a home region.

A home region is updated by sending a location update packet to the region, and the first location server to obtain the packet would carry out a broadcast in the region to update all location servers in that region of the movement of $u$. Multiple home regions are informed by location updates that traverse a multicast tree such that each update traverses a distance $K$ between two home regions. The length of such a tree is then $O(\frac{A}{R})$. Thus it is relatively easy to understand that all home regions know that $u$ is in $Q_{i'}$. In addition, all near home regions know that $u$ is in $R_{i'}$.

Discovering a node’s location: A node $v$ wishing to communicate to another node $u$ uses the mapping function to identify the closest home region of $u$ and sends a query packet to it. If the home region is a near home region, a response is generated, by the location server that gets the query, with $u$’s exact location. If the home region is a far home region, $v$ forwards it’s message to the closest near home region of $u$, and the location server that receives this message then forwards it to the exact location of $u$.

With the above, it can be shown that the worst case asymptotic location management cost for SLALoM is $O(vN^2)$, where $N$ is the number of nodes in the network, and $v$ is the average velocity with which a node moves. Having multiple home regions ensure that location discovery terminates deterministically in SLALoM. Also, since far home regions are updated less frequently, this helps to alleviate the volume of update traffic in SLALoM.

B. Description of ELF

The grid based topology construction is ELF is similar to that of SLALoM except that we do not have the concept of near or far home regions. Instead, we have forwarding home regions and a terminal home region. Thus, similar to SLALoM, there are $O(\frac{1}{K^2})$ home regions, but of which only one is a terminal home region, and the rest are forwarding
Function. There can be two cases:

- If the home region is a terminal home region, a location update cost is incurred. The average location update cost generated by SLALoM is proportional to $v$ and $N^{4/3}$. In this section, we shall show that, under the same assumptions, the average overhead cost of ELF is proportional to $v$ and $N^\theta$ (with $3/2 \leq \theta \leq 4/3$), and hence, performs no worse than SLALoM. The overhead cost of a location management protocol can be divided into three parts:

  - **Location update cost**: This cost covers all the messages nodes send to their home regions whenever they move to a new location.
  
  - **Location maintenance cost**: This cost covers all the messages nodes send to their previous Order-1 squares to inform them of their departure, and current Order-1 squares to inform them of their arrival and (c) collect as they are now location servers for the nodes currently registered in the new Order-1 square.
  
  - **Location discovery cost**: This cost covers all the messages sent for locating a mobile.

Mobility Model: In our mobility model, we assume that nodes move randomly and independent of each other. Each node selects a direction to move, chosen uniformly between $[0, 2\pi]$. Each node selects its speed, chosen uniformly between $[v - c, v + c]$ for some time $t$, where $t$ is distributed exponentially with mean $\tau$. After a mobile has travelled for time $t$, it selects another direction, speed and time to travel. As a consequence of this model, the average degree of a node will be proportional to $\pi r_v^2 N / A$ where $\pi r_v^2$ is the area within a node’s transmission range. To keep this fraction constant, $A$ must grow linearly with $N$.

For the grid based topologies considered, the original area is partitioned into unit regions. Based on the above mobility model, the size of the unit region is chosen so that its average node density is $\gamma$, a constant. Hence, there are $G = N / \gamma$ unit regions, each with area $a$. The main observations from [8] are the following:

1. The cost of broadcasting in an Order-1 square by a node is proportional to the number of transmissions needed to cover the said square. The latter is in turn proportional to the area of the Order-1 square divided by the area covered by a single transmission. Thus, $b = O(a/r_v^2)$ packets per Order-1 square.
2. The distance a node has to cover to cross an Order-1 square is proportional to the side of an Order-1 square. Thus, the number of Order-1 squares a node crosses per second, \( p_1 \), is proportional to \( v/\sqrt{a} \).

3. Given a source-destination distance \( d \), the number of transmissions that need to be carried out to send a packet from the source to destination is given by \( d/z \), where \( z \) is the average forward progress made in the course of one transmission.

Note that from [8], it is known that \( z \) can be computed from \( r_1 \) and the average degree of a node in the network. Also, by using a similar argument above, \( p_2 \), the number of Order-2 squares a node crosses per second, can also be estimated by

\[
O(p_1/K) = O(v/K\sqrt{a}) \text{ Order-1 squares per second, and } p_3, \text{ the rate at which a node updates all home regions is } O(p_1/\alpha K^{1+\beta}) = O(v/\alpha K^{1+\beta}\sqrt{a}) \text{ Order-1 squares per second.}
\]

**Location update cost:** Let us denote the location update cost per second per node as \( c_u \). Recall that each time a node moves into a new Order-1 square, it has to inform its terminal home region of its current exact location. This entails one broadcast in a unit region. Furthermore, if such a move also causes the node to move into a new Order-2 square, and the threshold is less than \( \alpha K^3 \) then it has to inform its previous terminal home region to set up a forwarding pointer to the current terminal home region. This again, requires another broadcast in a unit region. However, if the threshold is greater than \( \alpha K^3 \), then all home regions need to be notified of its current approximate location. This requires \( O(A/K^2) \) broadcasts in a unit region. Recall that the cost of updating all home regions is proportional to the sum of the length of all edges in the multicast tree spanning all home regions, which is proportional to \( O(A/K) \).

\[
c_u = O(p_1(b + K/z) + p_2(b + K/z) + p_3(A/K^2 b + A/K z)) \text{ packets/sec/node}
\]

Substituting \( p_1 \), \( p_2 \) and \( p_3 \) we have

\[
c_u = O(vK + vN/K^{3+2}) \text{ packets/sec/node} \tag{1}
\]

**Location maintenance cost:** Let \( c_m \) denote the location maintenance cost per second per node. The location maintenance phase consists of two primitives: (a) broadcast on arrival into a new \( L_0 \) grid and, (b) receive neighbor/location information to keep as criteria for being a member of the new grid. Hence,

\[
c_m = p_1 (b + \delta) \quad (1 \leq \delta \leq \gamma) = O(v) \text{ packets/sec/node} \tag{2}
\]

**Location discovery cost:** Let \( c_l \) denote the cost of locating a node per second per node. If \( v \) wishes to find the location of a node \( u \), it sends a location query to a home region of \( u \) closest to it. By construction, such a home region is at most \( O(K) \) away. If this is a terminal home region, then \( v \) obtains the exact location of \( u \). Thus, \( c_l = O(K/z) \) packets per second per node. On the other hand, if the home region is forwarding, then \( v \) obtains an approximate location of \( u \). Node \( v \) then routes its message to the home region of the last known Order-2 region of \( u \) (see Fig. 2). The message then follows the forwarding chain set up by \( u \) before it is handed to \( u \). In this case, the additional distance the message has to traverse

\[
d = x + y - z \leq 2y = O(y) = O(K^{\beta+1})
\]

Thus,

\[
c_l = O(K/z) + O(K^{\beta+1}/z) = O(K^{\beta+1}) \text{ packets/sec/node.} \tag{3}
\]

In this analysis, we shall make the assumption that packets arrive at each node at a rate of \( v \) packets/sec according to a Poisson process.

**Total Overhead Cost:** Combining the results above, we have the total overhead cost for the entire network. Thus

\[
c_t = N(c_u + c_m + c_l) = O(vNK + vN^2/K^{3+2} + vN + vN K^{\beta+1}) = O(vN^2/K^{3+2} + vN K^{\beta+1}) \tag{4}
\]

**Theorem.** The total overhead cost of ELF protocol is \( O(vN^\theta) \) \((4/3 \leq \theta \leq 3/2)\) packets per second.

**Proof:** Minimizing \( c_t \) with respect to \( K \), we have

\[
K = \left( \frac{\beta + 2}{\beta + 1} \right)^{3/4} N^{1/4}
\]

and

\[
c_t = O(vN^{3\beta+4}/2^{3+3}) \tag{5}
\]

Let \( \theta \) be \( 3\beta+4/2\beta+3 \). The lower bound for \( \theta \) results when \( \beta \) is zero and the upper bound results when \( \beta \) is a large number.
Simulation Time | 120 s
Simulation area | 1000x1000m - 3000x3000m
Unit($L_0$) Grid Size | 250 m
Number of Nodes | 80 - 720
Node Density | 80 nodes/km²
Transmission Range | 350 m
Transmission Speed | 2 Mbps
MAC Protocol | IEEE 802.11
Mobility Model | Random Waypoint
Maximum Speed | 10 m/s
Minimum Speed | 0 m/s
Pause Time | 0 s
CBR connections | 1000
Data Payload | 1024 bits
Traffic Pattern | Random

Fig. 3. Simulation Parameters

IV. SIMULATION RESULTS

While the worst case asymptotic performance is of theoretical importance, to compare the average case performance of each location management scheme, we implemented both ELF and SLALoM in Glomosim [11]. The simulation parameters are shown in figure 3.

We fixed the number of Order-1 grids per Order-2 regions to be four, and the length of the forwarding chain to be two($K = 2, \beta = 0, \alpha = 2$) for the simulations. Note that fixing $K$ to 2 brings out the worst case performance for each of the protocols. For each scenario, 1000 CBR connections were randomly generated, with each session sending a 1024 bit data payload. A session terminates successfully if the location discovery phase returns the correct location so that the data sent to the same location is successfully received by the destination before the simulation ends. We used MFR routing [8] for geographic forwarding, where packets are greedily forwarded by intermediate nodes to reach the intended recipient. Each scenario was ran for a period of 120 simulation seconds and each data point presented in the plots represents an average of five simulation runs.

Figure 4 shows the average signalling cost for both ELF and SLALoM. It is clearly visible from the figure that location update cost dominates other costs (discovery and data transfer) in both schemes, and that ELF performs much better than SLALoM. Since the number of full updates is lesser due to the forwarding chain set up, ELF is able to save on location update cost.

Figure 5 shows CBR session error probability. Due to the heavy signalling overhead, location discovery packets and data packets suffer heavy queueing under SLALoM, and hence fail to make it to the destination. It is quite remarkable that under heavy signalling, the delay experienced by packets is in the order of seconds (see fig. 7), and hence, there is a good chance that a location response from the location discovery phase would get dropped, since the originator of the query would have moved out from the grid where it had initiated the query phase. Overall, the number of successfully terminated sessions is much more for ELF than SLALoM.

Figure 6 shows the average path length (in hops) for data packets. Clearly, the average path length increases for both protocols with increase in terrain size. However, data has to be transmitted over additional hops due to the forwarding chain set up in ELF. On the average, this results in at most one hop more than that of SLALoM for the scenarios considered, and hence may not be a serious issue in practice. But data packets benefit indirectly from location forwarding, as can be seen from figure 7. As the number of nodes increase, so does the volume of update traffic, thereby increasing the network queueing delay at each node for session initiation and successful delivery of data for both protocols. The higher volume of update traffic causes congestion and excessively high packet delays in SLALoM, where as location forwarding results in a much better average delay experienced by both signalling and data packets in ELF. Even with location forwarding, the location discovery phase is carried out faster in ELF since location query/response packets reach the intended recipients with a lower delay. Due to network congestion, data packets suffer high delays in SLALoM, and with the increase in number of nodes, only those packets transmitted between source-destination pairs that are connected via few hops are successfully received (as indicated by the poor throughput and the average path length). Since we took into consideration the delay of successfully received data packets only, the delay of data packets in SLALoM start decreasing beyond a threshold, indicating a system breakdown. Overall, considering location update/discovery overhead/delay, and data delivery probability, ELF outperforms SLALoM for all practical purposes.

V. CONCLUSION

We have described novel schemes for location management in large ad hoc networks, and shown that the large volume of signalling traffic is a possible obstacle to achieving scalability.
in ad hoc networks. We have devised a new scheme known as ELF, which extends the idea of location forwarding from PCS networks, to curb the volume of update traffic in geographic routing based ad hoc networks. We find that under worst case analysis, ELF performs no worse than SLALoM, another location management scheme proposed in literature, and show by simulations that ELF performs much better than SLALoM for all practical purposes. All in all, ELF is a promising contender for location management for a wireless ad hoc network architecture.

REFERENCES


