PERFORMANCE EVALUATION OF TABLE-DRIVEN AND ON-DEMAND AD HOC ROUTING PROTOCOLS

Sung-Ju Lee
Computer Science Department
UCLA
Los Angeles, CA 90095
sjlee@cs.ucla.edu

Chai-Keong Toh
School of ECE
Georgia Tech
Atlanta, GA 30332
cktoh@ece.gatech.edu

Mario Gerla
Computer Science Department
UCLA
Los Angeles, CA 90095
gerla@cs.ucla.edu

Abstract—**Bandwidth and power constraints are the main concerns in current wireless networks because ad hoc mobile wireless networks rely on each node in the network to act as a router and packet forwarder. This dependency places bandwidth, power, and computation demands on mobile hosts which must be taken into account when choosing the best routing strategy. In recent years, protocols that build routes based "on demand" have been proposed. The major goal of on-demand routing protocols is to minimize control traffic overhead. In this paper, we perform a simulation and performance study on some routing protocols for ad hoc networks. Distributed Bellman-Ford, a traditional table-driven routing algorithm, is simulated to evaluate its performance in multihop wireless networks. In addition, two on-demand routing protocols (Dynamic Source Routing and Associativity-Based Routing) with distinctive route selection algorithms are simulated in a common environment to quantitatively measure and contrast their performance.**

I. INTRODUCTION

An *ad hoc network* is an infrastructureless network with no fixed routers, hosts, or wireless base stations. In such networks, a remote mobile node interconnection is achieved via peer level multihopping technique. This implies that the interconnection topology can change dynamically, giving rise to many challenging research issues. In this environment, ad hoc routing is critical and has to be supported before any applications can be deployed for ad hoc mobile networks. Routing protocols used in conventional wired networks (e.g., Bellman-Ford and link state) are not well-suited for the mobile environment due to the considerable overhead produced by periodic route update messages and their slow convergence to topological changes.

Numerous ad hoc routing protocols have been proposed to the Internet Engineering Task Force (IETF) Mobile Ad hoc Networks (Manet) Working Group [3]. Most of these protocols share the characteristic that routes are established based on demand by the source (hence the term *on-demand routing*). Some of these protocols have been evaluated via simulation, but in most cases they are not simulated in a common environment. This makes quantitative performance comparisons among these protocols difficult. In this paper, we use a common simulation platform to investigate the performance of three routing schemes. We compare Associativity-Based Routing (ABR) [7], which considers multiple route selection metrics, with Distributed Bellman-Ford (DBF), which is *not* an on-demand scheme, and Dynamic Source Routing (DSR) [4], which still uses shortest path as the routing metric. We have chosen these three protocols for the following reasons: (i) to evaluate the performance of a conventional table-driven routing scheme (DBF) in multihop wireless networks, and (ii) to study the performance of different routing metrics in dynamic ad hoc networks.

The remainder of this paper is organized as follows: Section 2 presents a discussion on DBF, DSR, and ABR protocols, highlighting how each of them supports route setup and mobility. Section 3 then evaluates and compares the performance of DBF, DSR, and ABR protocol via simulation. The conclusion follows in Section 4.

II. EXISTING AD HOC ROUTING PROTOCOLS

Recently, several proposals to support ad hoc mobile communications have evolved. In this paper, we focus our attention and discussion on DBF, DSR, and ABR.

A. Distributed Bellman-Ford

Distributed Bellman-Ford (DBF) algorithm was developed originally to support routing in the ARPANET. It is a table-driven routing protocol, i.e., each router constantly maintains an up-to-date routing table with information on how to reach all possible destinations in the network. For each entry, the next router to reach the destination and a distance to the destination are recorded. Each node in the network begins by informing its neighbors about its distance to all other nodes. The receiving nodes extract this information and modify their routing table if any route measure has changed. The node uses the following formula to calculate the best route:

$$D(i,j) = \min_{k}[d(i,k) + D(k,j)]$$

where $D(i,j)$ is the metric on the "shortest" path from node i to node j, $d(i,k)$ is the cost of traversing directly from node i to node k, and k is one of the neighbors of node i. After recomputing the metrics, nodes pass their own distance information to their neighbor nodes again. After a while, all nodes/routers in the network have a consistent routing table to all other nodes.

This protocol does not scale well to large networks due to a number of reasons. One problem is the so called "count-to-infinity" problem. In unfavorable circumstances, it takes up to N iterations to detect the fact that a node is disconnected, where N is the number of nodes in the network. Another problem is the increase of route update overhead with mobility. RIP uses time-triggered and event-triggered routing updates. In a mobile network environment, event-triggered routing updates tend to outnumber the time-triggered updates, leading to excessive overhead and inefficient usage of the limited wireless bandwidth.

B. Dynamic Source Routing

B.1 Protocol Characteristics. Dynamic Source Routing (DSR) [4] is a direct descendant of the source routing scheme used in bridged LANs. It uses source routing instead of hop-by-hop packet routing. Each data packet carries the list of routers in the path. The main benefit of source routing is that intermediate nodes need not keep route information because the path is explicitly specified in the data packet. DSR does not require any kind of periodic message to be sent, supports uni-directional and asymmetric links, and sets up routes based on demand by the source.

B.2 Route Discovery. When a source has a data packet to send but does not have any routing information to the destination, the source initiates a route discovery. To establish a route, the source floods a `route request` message with a unique request ID. When this request message reaches the destination or a node that has route information to the destination, it sends a `route reply` message containing path information back to the source. The "route cache" maintained at each node records routes the node has learned and overheard over time to reduce overhead generated by a route discovery phase.

When a node receives a `route request` packet, this message is forwarded only if all of the following conditions are met: (a) the node is not the target (destination) of the `route request` packet, (b) the node is not listed in source route, (c) the packet is not a duplicate, and (d) no route information to the target node is available in its route cache. If all are satisfied, it appends its identification to the source route and broadcasts the packet to its neighbors. If condition (b) or (c) is not met, it simply discards the packet. If a node is the destination of the packet or has route information to the destination, it builds and sends a `route reply` to the source, as described above.

B.3 Route Maintenance. DSR monitors the validity of existing routes based on the acknowledgments of data packets transmitted to neighboring nodes. This monitoring is achieved by passively listening for the transmission of the neighbor to the next hop or by setting a bit in a packet to request an explicit acknowledgment. When a node fails to receive an acknowledgment, a `route error` packet is sent to the original sender to invoke a new route discovery phase. Nodes that receive a `route error` message delete any route entry (from their route cache) which uses the broken link. Note that a `route error` message is propagated only when a node has a problem sending packets through that link. Although this selective propagation reduces control overhead (if no packets traverse a link), it yields a long delay when a packet needs to go through a new link.

C. Associativity-Based Routing

C.1 Protocol Characteristics. Associativity-Based Routing (ABR) [7] is a protocol that also builds routes on demand. The uniqueness of this scheme is the route selection criteria. By exploiting the spatial and temporal relationship of mobile hosts, ABR introduces the following new routing metrics:

- Longevity of a route based on associativity,
- Route relaying load of intermediate nodes supporting existing routes, and
- Link capacities of the selected route.

Associativity is measured by recording the number of beacons received by a node from its neighbors. For example, assume each mobile host has a radio range of $10m$ in diameter and there are two mobile hosts A and B. Initially, A and B are not in radio connectivity with each other but each sends a control beacon to signify its presence once every 2 seconds. If A is migrating at 1 m/s and it starts to enter B's radio range and move through it diagonally, both A and B record at most 5 beacons each. Hence, this is the associativity threshold. If only 5 or less beacons are recorded, then one can assume that the other mobile host is migrating past it, and this situation is viewed as being *associatively unstable.* Otherwise, if more than 5 beacons are recorded, the node is regarded as being *associatively stable.* By selecting nodes with high associativity counts, the route is expected to have a long-lived characteristic.

C.2 Route Discovery Phase. The route discovery process consists of Broadcast Query (BQ) and BQ-REPLY cycle. When a source demands a route, it floods a `BQ` message. If the `BQ` packet is not a duplicate, an intermediate node (IN) appends the following to the `BQ` packet: (a) its identifier, (b) associativity ticks with its neighbors, (c) route relaying load, (d) link propagation delay, and (e) hop count information. The IN then broadcasts the packet to its neighbors.

When the destination receives `BQ` packets, it knows all the possible routes and their qualities. The destination then selects the best route based on longevity and other qualities (route load, minimum hop, etc.) and sends a `BQ-REPLY` control packet (which contains a list of INs' addresses/IDs and a summary of selected route QoS) back to the source node via the selected route. When INs of the selected route receive the `BQ-REPLY` packet, they update their routing tables with this new route.

C.3 Route Reconstruction (RRC) Phase. When nodes' mobility invalidate the selected route, the Route Reconstruction (RRC) process is invoked to discover alternate partial routes quickly. The migration of neighbors can be detected when no beacon message is received within the timeout interval. When an IN of an existing route moves away from radio range of its immediate upstream or downstream, the route is invalidated. The immediate downstream node sends a Route Notification (`RN`) packet towards the destination to inform the invalidity of that route. Nodes that subsequently receive such a message delete their route entry. The immediate upstream of the moved node, however, performs a Localized Query (`LQ`) to discover a new partial route. Similar to `BQ`, information about route metrics is appended into `LQ` packets as they make their way to the destination. After the destination node receives several `LQ` messages, it selects the best partial route and sends back a `LQ-REPLY` message to the node that invoked the `LQ` process. As a result, all nodes in this partial path have their routing entry updated.

In the case when the node that sent the `LQ` message does not receive the `LQ-REPLY` message within the timeout period, it sends a `RN` packet to the immediate upstream node. When a node receives a `RN` packet from an immediate downstream node, it recognizes the backtrack and invokes a `LQ` process again. The fundamental strategy here is to *localize* the route discovery process to a bounded region so that other parts of the route are not affected. For a displacement of a node along the route, `LQ` processes can be performed at most half the route hop distance. Thereafter, if no partial path can be located, a RN message is sent back to the source node of the route to invoke a `BQ` process. This quick abort mechanism is to shorten route recovery time by limiting the number of `LQ` processes.

III. PERFORMANCE EVALUATION

A. Simulation Model

The simulator for evaluating three protocols is written using PARSEC [1]. The simulation models the network of 30 mobile hosts migrating within a $20m \times 20m$ space with a transmission radius of $5m$. Every node in the network moves in a random fashion, with a dormant time from 1 to 5 seconds before migrating again. The channel capacity is 2Mbps. A simple busy tone access scheme [6] is used as the medium access control protocol. Basically, a simplified channel propagation model is used. If nodes are within radio range of each other, they are considered connected and can exchange packets without errors if there are no collisions. A traffic generator was developed to simulate constant bit rate sources. Source nodes and destination nodes were chosen randomly with uniform probabilities. Simulation runs of 200 seconds of simulation time were performed multiple times.

B. Simulation Results

DBF, ABR and DSR are compared in a common multihop mobile wireless network simulation platform. Specifications stated in [5] and [2] are employed to implement DBF and DSR in our simulator in addition to ABR. The results obtained are discussed below.

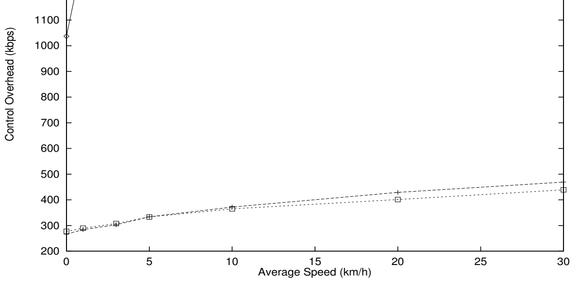

Fig. 1. *Control Message Overhead.*

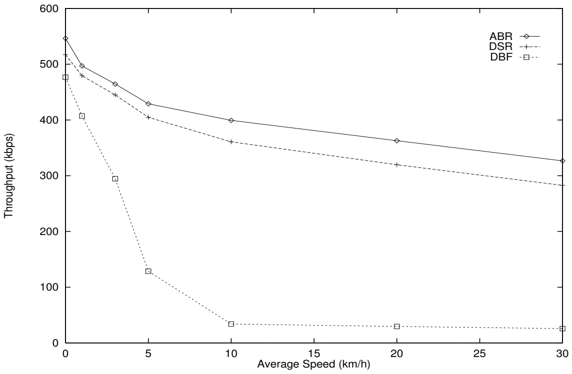

Fig. 2. *Data Throughput.*

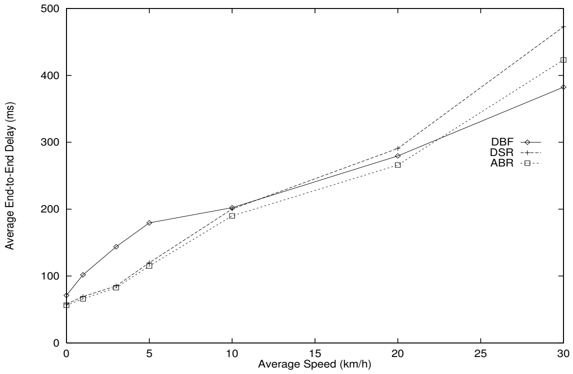

Fig. 3. *Average End-to-End Delay.*

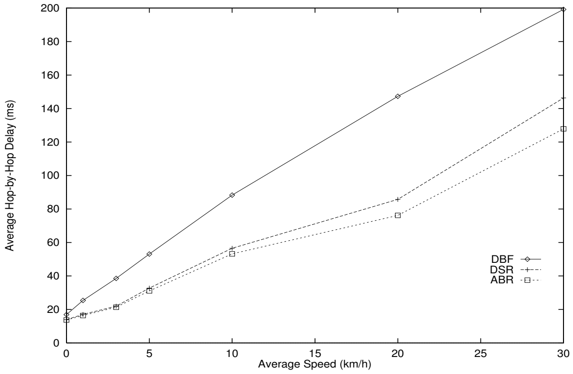

Fig. 4. *Average Hop-by-Hop Delay.*

B.1 Control Message Overhead. Figure 1 shows the control overhead incurred by DBF, DSR, and ABR. On-demand routing schemes have considerably less overhead (as high as 78.91%) than DBF. Sending route updates periodically and triggering updates when the topology changes result in excessive control message overhead, which is unacceptable in a wireless environment with limited bandwidth. We can see that DSR has less overhead than ABR when the network is static. If nodes are not mobile, there is no route breakage and control messages for route reconstruction are not required. ABR sends beacon messages to maintain the list of neighbors, thus resulting in more overhead when there is no mobility. One might expect ABR to have considerably more control overhead when nodes are stable. However, the result shows only a small difference since we have calculated the route header in each data packet as control overhead. Remember that ABR lists only previous and next hop while DSR lists the entire path in data packets.

We can observe from the result that increasing the mobility speed makes ABR more efficient than DSR. This efficiency is attributed to ABR's local route recovery feature. In DSR, if a node in the path becomes unreachable, a control message specifying a route error is propagated all the way back to the source to invoke a new route discovery. In contrast, in ABR the immediate upstream of a migrated node starts the `LQ` process to find a new partial route without intervention from the source, hence minimizing the transmission of control messages.

B.2 Data Throughput. Figure 2 shows the throughput comparison of DBF, DSR, and ABR. DBF's poor performance can be attributed to excessive channel usage by route update control messages. Also, as mobility speed increases, more event-triggered updates are generated. However, this is not present in on-demand routing protocols. The graph also reveals that the ABR has a higher throughput than DSR, resulting from the use of a different route selection process. In DSR, a route is chosen based on the shortest delay at the instance of route establishment. Although this path may be the best route at that instant, it may be a route that lacks routing stability or may have unacceptably high load. In contrast, ABR distinctively selects a route where nodes in the path are associatively stable (spatial, temporal, and connection wise) and have light load. This route selection criteria enhances the longevity of the selected route, avoids bottleneck and congestion at INs, and eventually improves throughput.

B.3 End-to-End and Hop-by-Hop Delay. Figure 3 and 4 show the end-to-end and hop-by-hop delay of data packets respectively. Note that DBF is comparable with other protocols at high mobility speed in Figure 3, because delay is measured for packets that have survived to reach the destination, and in DBF most packets are for shorter-hop routes in high mobility. However, Figure 4 reveals that DBF has a larger hop-by-hop delay than on-demand schemes due to high control overhead and thus large queueing delay. For on-demand protocols, ABR has shorter de-

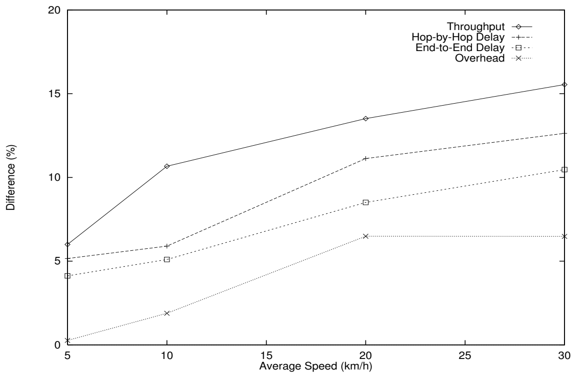

Fig. 5. *Summary Comparison of ABR vs. DSR.*

lays than DSR, and this difference becomes more obvious as mobility speed increases. The better performance of ABR can be traced to the following reasons. First, balancing the route load shortens the delay as the chance of congestion is reduced. Second, adjusting to network mobility via receiving beacon messages from neighbors yields faster convergence. In DSR, a neighbor displacement is noticed only after a packet is sent explicitly to that node. The network reacts if an acknowledgment is not received. Consequently, this increases packet delay since the packet must wait until a new route is established.

B.4 Summary Comparison of ABR and DSR. To provide a summary comparison of ABR and DSR, we report the absolute value of the difference for various measures in Figure 5. The values are calculated using the following formula:[1]

$$\frac{|value_{ABR} - value_{DSR}|}{value_{DSR}} \times 100(\%)$$

ABR is superior to DSR over all measures as the percentage difference represents the improvement of ABR over DSR. We can see that ABR gives better throughput, less control overhead, and smaller delay than DSR. The percentage difference grows as the mobility increases.

IV. CONCLUSIONS

Many routing protocols for ad hoc mobile wireless networks have been proposed in recent years. In this paper, we have reviewed and studied key properties of three distinctive routing protocols. Performance evaluation of these protocols have been conducted via simulation in a common network environment. We have compared the performance of Associativity-Based Routing with Distributed Bellman-Ford and Dynamic Source Routing. Simulation results reveal that the DBF incurs extensive bandwidth and computation overhead in the presence of mobility, yielding inferior performance when compared to on-demand routing protocols (ABR and DSR) in ad hoc networks. We also report that ABR has a better throughput, smaller delay, and lower control overhead than DSR. Chiefly, this is due to the use of innovative associativity criterion, multiple route selection metrics, and local route recovery.

In summary, ABR is a strong candidate for the multihop mobile wireless environment along with DSR. The final selection of the on-demand routing scheme should take into account other considerations in addition to the measures provided by simulation.

References

[1] 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.

[2] J. Broch, D.B. Johnson, and D.A. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," *Internet Draft*, `draft-ietf-manet-dsr-00.txt`, Mar. 1998. Work in progress.

[3] Internet Engineering Task Force (IETF) Mobile Ad Hoc Networks (Manet) Working Group Charter.

[4] D.B. Johnson and D.A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," In *Mobile Computing*, edited by Tomasz Imielinski and Hank Korth, Chapter 5, Kluwer Academic Publishers, 1996, pp. 153-181.

[5] G. Malkin, "RIP Version 2 - Carrying Additional Information," *Internet Draft*, `draft-ietf-ripv2-protocol-v2-05.txt`, Jun. 1998. Work in progress.

[6] F.A. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part II-The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution," *IEEE Transactions on Communications*, vol. COM-23, no. 12, Dec. 1975, pp. 1417-1433.

[7] 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.

---

[1] Note that we use absolute values since the value of ABR is higher than DSR in throughput, but smaller in overhead and delay.