Attachment 1a: Copy of Document 1 from the Northwestern University Library

Northwestern University Interlibrary Loan

TN: 818262

Borrower: UIU

Lending String:
EEM,*INU,NJR,CLU,NRC,INT,JHE,CIN,FHM,CUV,
LHL,AUM,PUL

Patron:

Journal Title: 1997 IEEE International Conference on Communications : towards the knowledge millennium : ICC '97, 8-12 June 1997, Montréal, Québec, Canada, confere

Volume: 1997 (this has 3 volumes, from **Issue:**
Month/Year: 1997**Pages:** 1381-1386

Article Author: Shoubridge, Peter and Arek Dadej

Article Title: Hybrid Routing in Dynamic Networks; Also table of contents and cover page that shows acquisitions date

Imprint: [New York] : Institute of Electrical and Electronics Engineers ; Piscataway, NJ. : IEEE Service Center, ©1997.

Barcode:

35556027685031 35556027685023 3555602768 5015

Location: Oak Grove Library Center

Charge
Maxcost: 35.00IFM

Shipping Address:
Interlibrary Borrowing
128 Library University of Illinois
1408 West Gregory Drive
Urbana, Illinois 61801
United States

EXHIBIT

Ex. 1005

Hybrid Routing in Dynamic Networks

Peter Shoubridge
Communications Division, Defence Science Technology Organisation
Salisbury, South Australia, 5108

Arek Dadej
Institute of Telecommunications Research, University of South Australia
The Levels, South Australia, 5095

Abstract—**In mobile radio communication networks the distribution of traffic loads and network topologies may vary from nearly static to very dynamic. This dynamic behaviour may vary both in space and in time. Since routing algorithms tend to be well suited to specific networking environments, it is very difficult to select a single routing algorithm that is most appropriate for a given network, if the network is subjected to varying degrees of dynamic behaviour. This paper proposes a routing strategy that smoothly adapts to changing network conditions by combining two distinct routing principles into a single hybrid routing procedure. The hybrid routing strategy exhibits a smooth change from shortest path routing to constrained flooding, as the behaviour of the network (or regions within) changes from quasi-static to very dynamic.**

I. INTRODUCTION

In addition to successfully forwarding user traffic between source and destination nodes in a communications network, the routing function generally seeks to optimise some performance criteria such as network utilisation or throughput, or perhaps average delay. As networks increase in size, with more users demanding seamless connectivity, mobility and a high level of availability, the task of routing becomes an increasingly complex problem to network providers.

For efficient routing, nodes require knowledge of network topology. Links may fail or become congested, nodes or users could be mobile, resulting in changes to source-destination paths. Information regarding these changes must be conveyed to network nodes so that appropriate routing can take place. This information exchange consumes network resources and presents a cost in maintaining up to date routing information in network nodes. As topology or traffic loads change more frequently, the network becomes dynamic and maintaining accurate routing information at individual nodes comes at a higher cost. Inaccurate routing can result in lost or delayed traffic causing end-to-end retransmissions which further load the network. This additional load is itself an unwanted overhead which could be avoided with a better routing strategy.

Algorithms that can determine shortest paths between source and destination nodes are typically used in static or quasi-static networks. They require periods of network stability so routing tables can settle to reflect true shortest paths. The most efficient algorithm is that described by Dijkstra which requires complete knowledge of the network topology [1, p. 103]. This algorithm can be executed centrally or topology information can be broadcast to all nodes in a decentralised implementation [2]. Another algorithm, suited to distributed architectures, is a decentralised version of Ford and Fulkerson's algorithm which only requires information exchange between immediate neighbouring nodes [3, p. 268], [4, p. 130]. More recent algorithms derived from these offer significant improvements in performance [5], [6], [7].

Routing by flooding is often used in very dynamic networks where destination nodes are highly mobile or topological changes occur very frequently and therefore the network connectivity is uncertain. Flooding algorithms simply broadcast user traffic through a network ensuring that the destination will be reached because all paths to the destination are attempted. No routing tables or routing information exchange is required when using flooding algorithms, but flooding is very wasteful of network resources.

II. ROUTING COSTS

Routing algorithms consume network resources in determining routes through a network to the required destination. In addition to overheads such as processing time for algorithm execution and memory to store routing tables, there is the consumption of transmission capacity that could otherwise be utilised for user traffic. Typically, computer processing power and memory requirements within network nodes are of low relative cost when compared with transmission overhead requirements in networks using low to moderate link capacities.

Shortest path algorithms maintaining routing look up tables in each node contribute zero transmission cost when making a routing decision. However, they do consume resources every time the network changes by bringing all routing tables throughout the network up to date. If routing decisions are based on inaccurate information stored in routing tables, long delays or lost traffic can lead to retransmissions or reattempts caused by higher layer end-to-end protocols.

Flooding based routing procedures do not maintain routing tables and therefore do not require any table update procedures. As such they contribute zero update cost but do consume large amounts of network capacity in search of destinations each time a routing decision is required. Overheads resulting from routing decisions are generally independent from the frequency of changes occurring in networks.

While shortest path algorithms are less costly to operate in a quasi-static network environment, flooding may actually consume less resources in a very dynamic network. This is indeed true if the shortest path algorithm is unable to maintain accurate routing tables due to high rates of change in link cost and network topology.

In considering the overall routing costs associated with routing procedures such as these, it is possible to identify conditions where frequency of change within a network requires the use of one particular algorithm over another. Furthermore, closely examining routing algorithm performance under dynamic conditions highlights the more dominant causes of routing cost. Strategies can then be conceived to improve performance by addressing these more dominating effects.

III. ROUTING PERFORMANCE

Routing algorithm performance alone can be expressed in terms of convergence speed and routing table update message overhead. Further insight into overall routing procedure performance can be achieved by incorporating effects such as dynamic topologies, high network loads and end user retransmissions. Due to the difficulty in analysing adaptive distributed routing procedures [8, p. 318], [9, p. 211] and the desire to also include these additional conditions, computer simulation modelling has been used for the evaluation of routing strategies in this paper.

Transmission bearer capacity limits the total available resources within a network resulting in a threshold to average delay performance as network loads increase. As the average rate of user traffic load (γ) entering a network increases towards the saturation load (γ^*), overall average delay (T) departs from a "no load" delay with $T \to \infty$ at a traffic load of γ^* [8, p. 323]. If saturation load is approached and congestion is occurring, flow control mechanisms must be employed to relieve the congestion. A more efficient routing procedure will enable an increase in a given network's saturation load.

Using saturation load to assess a routing algorithm's performance is somewhat unrealistic because a network cannot be operated with infinite average delay. A more realistic measure is the maximum operating point, denoted γ', which is the maximum user traffic load entering the network before average delay departs from the "no load" delay and begins to grow rapidly. This is in effect just before the threshold "knee" of a delay-throughput curve. Maximum operating point γ' can be defined as the point where a line from the origin is tangent to the delay-throughput curve [10, p. 9]. Note that γ' as defined here is not true throughput but the maximum originating user traffic arrival rate. True throughput is required for the accurate comparison of routing procedure performance, and as described later this is derived from the simulation results.

A. Simulation model

An initial network model has been developed based on stationary stochastic processes evenly distributed across the network. This reduces bias and transient effects in simulation results. Consider the network model as a directed graph G with N nodes and L bidirectional links. Each node functions as a source of user traffic entering the network where traffic can be destined to all other nodes within the network. Routing algorithms are executed within these same nodes and update control messages are exchanged between neighbouring nodes for the purpose of maintaining routing tables. Routing is used to effectively provide a connectionless datagram service in forwarding user traffic through the network.

An adaptive distributed shortest path algorithm is selected for good performance when the network is static or quasi-static. This distributed algorithm offers higher survivability and if link cost functions reflect both connectivity and delay (node output queue lengths), minimum delay routing can be employed to enhance performance. However, incorrect congestion information leading to the choice of inferior routes, typically, has less serious consequences than incorrect topological information resulting in the possible choice of non-existent routes [11, p. 342]. A minimum hop shortest path algorithm has therefore been adopted where link costs are set to a constant value simply representing topology connectivity. The minimum hop algorithm reduces complexity in this investigation while still concentrating on the important issue of topological information update.

Routing algorithms are executed in each of the N nodes in G. Distance and routing tables are assumed initially configured with all known destinations and neighbouring nodes by some higher level topology control entity. In the actual simulation this occurs during initialisation.

Since the routing procedures are triggered by link cost change events, G has essentially fixed topology with links always existing between nodes and their respective neighbours. Topology change is modelled by having link quality changing between one of two possible states; a link is either up providing good error free transmission, or a link has failed providing no useful transmission at all. This simplified approach requires no topology control as a particular node's neighbours at any given time always belong to the same subset of nodes as originally defined in G.

The minimum hop routing procedure has been developed from the Jaffe-Moss distributed shortest path algorithm [6]. This algorithm has the advantage that it exhibits rapid convergence and is loop free. In addition, this algorithm can be used for more general least cost routing. To execute the algorithm as minimum hop, link costs are set to either the value 1 (link present) or ∞ (link failed). The Jaffe-Moss algorithm consists of two table update procedures. A link cost decrease causes an Independent Update Procedure (IUP) based on a decentralised version of Ford-Fulkerson's algorithm. Jaffe and Moss determined that if link costs decrease then algorithms such as Ford and Fulkerson's are inherently loop free. If a link cost increases, it is possible that loops may occur while the algorithm converges. To prevent the formation of loops a Coordinated Update Procedure (CUP) is used whenever link costs increase.

Routing table update messages used for the Independent and Coordinated Update Procedures are set at a fixed size of 8 octets (64 bits); this is considered sufficient to carry required information such as message type, source address, destination address, routing costs and other quality of service parameters. In the case of multiple new minimum costs occurring in distance tables, a random selection is made of nodes with the same distance to avoid any deterministic bias in the selection of a new next node.

Flood search routing has been selected for its robustness in dynamic networks and is modelled as constrained flooding, the most efficient way to flood an entire network [12]. Any

user packet transmitted from a node is copied and broadcast on all outgoing links. Intermediate transit nodes do not broadcast a packet on the same link that a packet was originally received on. Constrained flooding uniquely identifies packets associated with a particular flood search by using sequence numbering. Nodes store sequence numbers of packets already flooded. If any packets revisit a node with the same sequence number, they are discarded instead of being further broadcast to neighbours. This technique ensures that all nodes are visited at least once and duplicated traffic is kept to a minimum throughout the network.

A 64 node network with connectivity of degree 4 is modelled as G. The network is a large regular graph forming a manhattan grid network that has been wrapped around itself as a torus to avoid edge effects. Transmission links function as a single server queueing system with service rate defined by packet size and link capacity. Link capacity is fixed at 16kbps to represent a narrowband radio link or perhaps a logical signalling channel. Infinite length queues are modelled to ensure that packet discards result solely from routing table uncertainty, loop formation, and link failure.

User packets entering the network arrive with Poisson arrival rate γ_n at each node n. Each user packet entering the network at node n is randomly assigned a destination node d such that $d \in G$ and $d \neq n$. The total load entering (and leaving) the network, $\gamma = \sum_{n=1}^{N} \gamma_n$, is evenly distributed across all N nodes. User packet length is fixed and set to 128 octets (1024 bits). This is based on the default X.25 Packet Layer Protocol packet size [13, p. 354]. X.25 is typically used in wide area networks and modified versions have been proposed for packet radio networks [14], [15]. User packet size has been set larger than routing table update packet size to realistically represent data packets while effectively capturing the overheads caused by packet duplication when flooding.

Discarding of user packets due to routing table uncertainty creates a problem because the probability of successfully reaching a destination must now be incorporated into the analysis of results. Generally, in real systems lost traffic leads to reattempts or retransmissions by higher layer protocols. These retransmissions will inevitably place additional load on the network and be reflected as a transmission overhead cost. To include this behaviour and provide a more realistic model, a selective repeat Automatic Repeat reQuest (ARQ) protocol is modelled on an end-to-end basis between source-destination pairs with a retransmission timeout fixed at five seconds.

Destination nodes notify the source that a user packet has successfully been received by returning an acknowledgment packet. On receiving an acknowledgment, the source node cancels the retransmission timer for that outstanding user packet. A transport layer packet identifier is required for source nodes to uniquely identify originated packets being acknowledged. Transport layer retransmission acknowledgment packets are set to a packet size of 8 octets. This is to reflect the control functionality of this packet type as distinct to packets carrying user information. In a real system this information may be piggybacked on return user packets. The constrained flooding model does not utilise this acknowledgment procedure because of the high probability of successful transmission using this scheme.

Changes in network topology are evenly distributed across all links. The amount of change occurring in the network at any given time is denoted intensity of change, $\mathcal{I}$, and defined as the average number of link cost changes per second, per link. To reflect some stochastic behaviour in the simulation model, link cost change events are scheduled with event times derived from two negative exponential probability distributions. These distributions are characterised by two different mean interarrival times; one representing the mean interarrival time between link failures in the network and the other, average link down time. If a link failure event is scheduled to occur, a link (i, j) is randomly selected from the L possible links in G, using a uniform distribution. If the current state of the chosen link (i, j) is operational, then the link is failed. If the link (i, j) is already in a failed state, an alternative link is randomly selected repeating the process until an operational link is found. Both nodes i and j adjacent to the link are notified of the change in link cost. The link (i, j) is then placed in a failed state, queues are emptied and the link's restoration time is derived from the link restoration probability distribution and scheduled on a simulation event list.

The ratio of mean link failure interarrival time to mean link down time is regulated to ensure that on average only a given number of links are failed at any time. For the network simulation model, the average proportion of links failed at any time is 5%. This represents reasonable change to network topology without causing significant network partitioning. While the routing procedures can function in partitioned networks, gaining insight into the algorithm's behaviour is difficult. A link cost change event triggers routing procedures in nodes i and j to act upon the cost change for all affected destinations. Note that a link (i, j) refers to a bidirectional link connecting two neighbouring nodes so there are two queueing systems associated with each bidirectional link. When a link fails or recovers, the cost change simultaneously affects both directions on a single bidirectional link.

B. Simulation results

Performance of the constrained flooding and minimum hop algorithms using a static network model ($\mathcal{I} = 0$) yielded the results shown in Figure 1. Overall average delay experienced by all user packets is presented as a function of the scaled originating user traffic arrival rate per node. Each point on the curve represents average results from 5 simulation runs using different random number seeds. Confidence intervals of 95% are calculated from the t distribution. Validation was achieved by thorough testing of algorithm execution, accuracy of routing tables and packet tracing. Further verification of the simulation model was obtained by comparison of simulation results with the results derived from an M/M/1 queueing model. For this comparison user packet size was derived from a negative exponential probability distribution and the selective repeat ARQ protocol was disabled. These

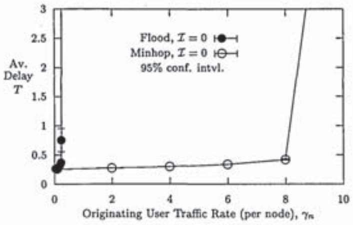

Fig. 1. Flooding and minimum hop in static network

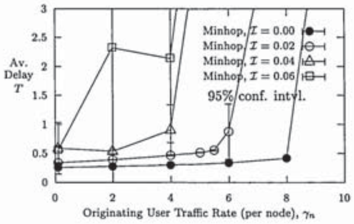

Fig. 2. Minimum hop with increasing change intensity

verification test results have not been reproduced here.

Duplicate packet overheads are the cause of reduced maximum operating point in constrained flooding. The high routing cost associated with flooding is due to searching the entire network every time a routing decision is required. The minimum hop algorithm requires no network resources for routing table updates in a static network, $\mathcal{I} = 0$. This results in a significantly higher maximum operating point. Characteristic threshold behaviour of both curves in Figure 1 is consistent with the behaviour of a network of M/M/1 queues.

Consider now performance of minimum hop routing when operating in a dynamic network environment. Figure 2 illustrates delay vs. throughput performance for intensity of change $\mathcal{I} = 0.0, 0.02, 0.04$ and 0.06. Performance degrades as the intensity of change increases. More network resources are consumed for maintaining the routing tables while changes in topology are being tracked. As routing table update costs increase with higher intensities of change, the network capacity available for user traffic is reduced.

If intensity of change is increased to $\mathcal{I} = 0.06$, the minimum hop algorithm no longer exhibits the expected threshold behaviour, see Figure 2. The increase in average delay and high variance in output statistic samples are due to the high rate of instances where next node entries are unknown, causing packet retransmissions. On average, many user packets are requiring retransmissions in order to successfully reach their respective destinations. The retransmission timeout delay is significantly affecting the average delay performance such that overall performance is being dictated by the performance of a higher layer "transport" protocol. The routing procedure is becoming ineffective under such dynamic conditions.

For constrained flooding, the maximum operating point remained the same, with $\gamma'_n = 0.2$, as $\mathcal{I}$ increased. As expected, constrained flooding is inefficient but independent from changes in network topology. It is reasonable to conclude that a large network similar to the one modelled, would require a flooding procedure if the network is to operate in a very dynamic, or potentially very dynamic environment. Unfortunately, flooding results in very low network utilisation and therefore better routing strategies need to be sought.

IV. HYBRID ROUTING

Many routing algorithms operate efficiently only for one class of networks (either static or very dynamic). The task of selecting a most appropriate routing algorithm can be very difficult because the dynamic nature of a given network can vary. It is possible to achieve improved performance over a wider range of network conditions by combining different routing procedures into a new hybrid routing procedure.

Somewhat similar in concept is a proposal to support user mobility in deployed circuit switched trunk networks [16]. With this scheme a new call request packet is forwarded using look up tables to a node servicing the destination user. If the destination user is no longer affiliated with that node, a flood search is initiated. Problems arise though with such an approach when networks exhibit dynamic topologies [17].

In observing the minimum hop routing procedure, it has been established that operation of the Jaffe-Moss algorithm in very dynamic networks is limited significantly by routing table uncertainty. While table update protocol overheads are substantial, they are less dominant. This observation opens an opportunity for significant improvements to the algorithm's performance by overcoming the problem of uncertainty in routing tables. As we will see from the results obtained, the proposed solution (simple local broadcast) leads to a routing procedure that exhibits a smooth change to flooding as intensity of change in the network increases.

A. Hybrid routing model

A hybrid routing strategy is now proposed where a local broadcast, derived from constrained flooding, is initiated if routing tables cannot provide a next node entry for forwarding user traffic. Under normal circumstances the Jaffe-Moss algorithm provides a next node entry for routing decisions. If this entry is unknown due to link failure, user traffic is broadcast on all outgoing links (except the incoming link). Subsequent nodes further downtree towards the destination may use minimum hop routing tables if a valid next node entry exists. All nodes remain involved in table update procedures. Once a node initiating a CUP has been acknowledged and a valid next node determined, local broadcasting is no longer used. It is worth observing that the routing procedure defined here automatically adapts to changes in the network. Flooding occurs only for the parts of the network where routing tables are uncertain and only for the periods of time when the uncertainty persists.

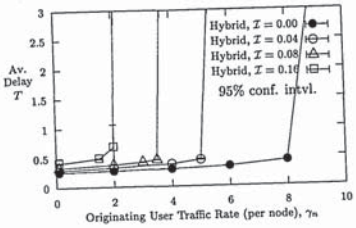

Fig. 3. Local broadcast hybrid routing with increasing change intensity

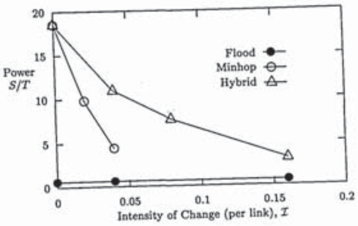

Fig. 4. Comparative performance over varying change intensity

Sequence numbering of packets is required in hybrid routing to manage the discarding of duplicated packets, as was done with constrained flooding. This use of sequence numbering also ensures that any packet reaching a destination does so along a loop free path because any packet with the same sequence number visiting a node more than once is discarded. While packets have a greater chance of reaching the destination due to multiple packets travelling along different paths, there is no guarantee that at least one packet will reach the destination. This is due to possible looping resulting in all packets being discarded. However, probability of reaching the destination is very high.

B. Performance of hybrid routing

Figure 3 demonstrates performance of the hybrid routing procedure for various intensities of change. Even with very high levels of change intensity, the hybrid scheme still exhibits threshold type behaviour. Throughput decreases as the intensity of change increases though due to the routing table update traffic and the flooded user traffic. Overall the effect of hybrid routing is a smooth transition from a shortest path routing procedure in static and quasi-static conditions towards a flooding procedure as the network becomes more dynamic. The efficiency of this scheme offers very significant improvements of throughput as compared to pure constrained flooding.

In using hybrid routing, network capacity is consumed by routing table update traffic and duplicate user and acknowledgment packets caused by local broadcasting. These overheads reduce available throughput for user traffic and represent the routing cost of this routing procedure. With minimum hop routing, overheads are dominated by routing table update traffic and user traffic retransmissions. Given that update traffic load will be the same for hybrid routing as it is for minimum hop routing, due to identical network and change conditions, it can be concluded that local broadcasting overheads are less than minimum hop retransmission overheads, resulting in the lower routing costs.

Comparing overall performance of the three routing procedures requires assessment of maximum operating points under varying dynamic conditions. True maximum operating point is a function of throughput (S) and delay (T), with the actual throughput calculated using Equ. (1). Maximum average throughput (scaled on a per node basis) is defined as the actual amount of traffic successfully delivered to destination nodes when the network is operating at its maximum operating point for a given intensity of change. Values for probability of successful transmission for node n, p_n, are measured by monitoring the proportion of packets received to those attempted to be sent. Routing power is defined as S/T and used to reflect the maximum operating point as a single parameter [10, p. 8].

$$S = \frac{\sum_{n=1}^{N} \gamma'_n p_n}{N} \tag{1}$$

Results of routing power vs. intensity of change for constrained flooding, minimum hop and hybrid routing are plotted in Figure 4. Flooding demonstrates poor efficiency but is robust in dynamic topologies. The relative inefficiency of flooding can be attributed to the ratio of user packet to control packet size of 16:1. Minimum hop routing degrades until it essentially fails as indicated in Figure 2. Hybrid routing clearly offers the highest maximum operating throughput over the range of static through to dynamic network conditions. Operation with high throughput in very dynamic conditions provides an indication of the algorithm's robust nature.

Desirable properties of the Jaffe-Moss shortest path algorithm and constrained flooding are preserved in this hybrid routing procedure. Rapid convergence, minimal update traffic load and loop free routes to the destination are guaranteed through the use of sequence numbering, while local broadcasting also provides a high assurance of successful transmis sion under very dynamic conditions. Disadvantages include the extra complexity and storage required within network nodes to manage sequence numbering of user packets. Also, increased packet overhead to contain the sequence numbers may be a disadvantage. These disadvantages, however, are outweighed by the improved throughput/delay performance and adaptive properties of the algorithm.

V. CONCLUSIONS

In closely investigating the performance of minimum hop routing, it was found that routing table update traffic contributes only a proportion of total overhead traffic loads. A more dominant factor in degrading network performance is caused by user end-to-end retransmissions. As rates of link cost change within the network increase, so does the extent of routing table uncertainty. This leads to high levels of lost user traffic and reflects the algorithm's inability to track high rates of topology change.

Hybrid routing exploits flooding characteristics using a local broadcasting strategy to address the routing table uncertainty problem and provides improved overall performance. The focus of this strategy is to incorporate complimentary advantageous features of generalised least cost and flooding type algorithms into a more flexible routing strategy capable of adapting to changing operational conditions. The resulting hybrid scheme is relatively simple, robust and very efficient in dynamic networking environments.

REFERENCES

[1] W-K. Chen, *Theory Of Nets: Flows In Networks*, John Wiley and Sons, 1990.

[2] J.M. McQuillan, I. Richer, and E.C. Rosen, "The new routing algorithm for the ARPANET," *IEEE Transactions on Communications*, vol. COM-28, no. 5, pp. 711–719, May 1980.

[3] M. Schwartz, *Telecommunication Networks: Protocols, Modeling and Analysis*, Addison-Wesley, 1988.

[4] L. Ford and D. Fulkerson, *Flows in Networks*, Princeton University Press, 1962.

[5] P.M. Merlin and A. Segall, "A fail safe distributed routing protocol," *IEEE Transactions on Communications*, vol. COM-27, no. 9, pp. 1280–1287, September 1979.

[6] J.M. Jaffe and F.H. Moss, "A responsive distributed routing algorithm for computer networks," *IEEE Transactions on Communications*, vol. COM-30, no. 7, pp. 1758–1762, July 1982.

[7] J.J. Garcia-Luna-Aceves, "Loop free routing using diffusing computations," *IEEE/ACM Transactions on Networking*, vol. 1, no. 1, pp. 130–141, February 1993.

[8] L. Kleinrock, *Queueing Systems Volume II: Computer Applications*, John Wiley and Sons, 1976.

[9] R.F. Garzia and M.R. Garzia, *Network Modeling, Simulation, and Analysis*, Marcel Dekker, Inc., 1990.

[10] L. Kleinrock, "Performance evaluation of distributed computer-communication systems," in *Queueing Theory and its Applications*, O.J. Boxma and R. Syski, Eds., pp. 1–57. Elsevier Science Publishers B.V. (North-Holland), 1988.

[11] D. Bertsekas and R. Gallager, *Data Networks*, Prentice Hall, 1987.

[12] T.V. Vu and S.P. Risner, "Performance and cost of broadcast routing algorithms in the strategic defense system terrestrial network," in *IEEE MILCOM*, 1991, pp. 29.2.1–29.2.5.

[13] A.S. Tanenbaum, *Computer Networks*, Prentice-Hall International, second edition, 1989.

[14] B.H. Davies, S.J. Swain, and K. Watkins, "Efficient implementation of X.25 on narrowband packet radio systems," in *IEE Colloquium on "Military Communications - The Trend Towards Civil Standards", (Digest No. 55)*, April 1990, pp. 6/1–6/5.

[15] P.R. Karn, H.E. Price, and R.J. Diersing, "Packet radio in the amateur service," *IEEE Journal on Selected Areas in Communi-*

[16] V.O.K. Li and R-F Chang, "Proposed routing algorithms for the U.S. Army mobile subscriber equipment (MSE) network," in *IEEE MILCOM*, 1986, pp. 39.4.1–39.4.7.

[17] L.E. Miller, R.H. French, J.S. Lee, and D.J. Torrieri, "MSE routing algorithm comparison," in *IEEE MILCOM*, 1989, pp. 2.2.1–2.2.5.