United States Patent [19]

Vu

[11] **Patent Number: 5,056,085**

[45] **Date of Patent: Oct. 8, 1991**

[54] **FLOOD-AND-FORWARD ROUTING FOR BROADCAST PACKETS IN PACKET SWITCHING NETWORKS**

[75] Inventor: **Thu V. Vu**, West Melbourne, Fla.

[73] Assignee: **Harris Corporation**, Melbourne, Fla.

[21] Appl. No.: **391,197**

[22] Filed: **Aug. 9, 1989**

[51] **Int. Cl.**[5] **H04Q 11/04**

[52] **U.S. Cl.** **370/60**; 370/94.1

[58] **Field of Search** 370/60, 60.1, 94.1, 370/94.2, 94.3, 85.13, 85.14

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,399,531 8/1983 Grande et al. 370/60
4,905,233 2/1990 Cain et al. 370/94.1

OTHER PUBLICATIONS

Computer Networks, by Andrew S. Tananbaum, Prentice Hall, Englewood Cliffs, N.J., 1981.

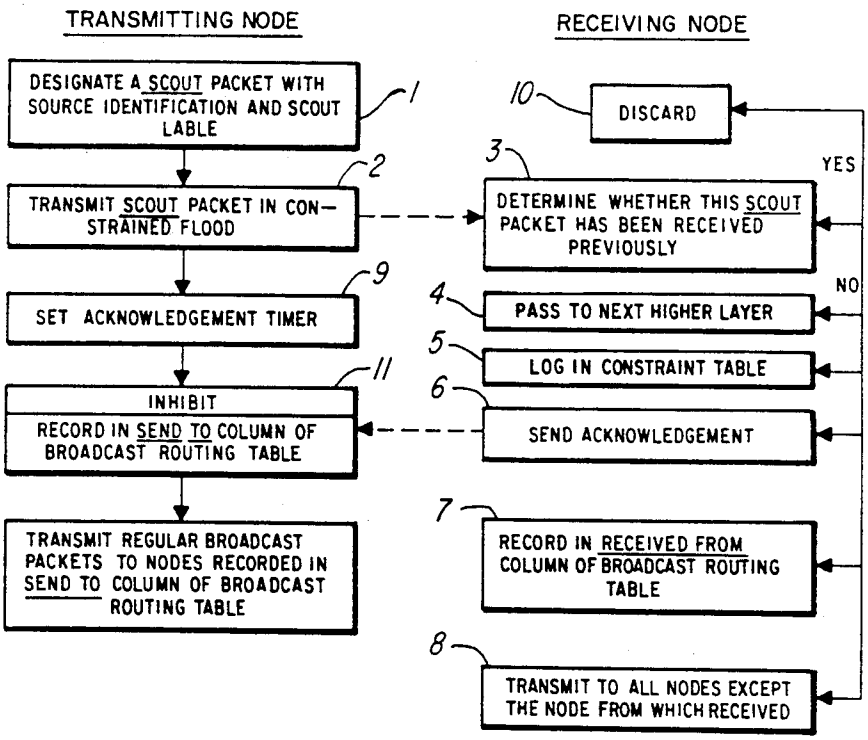
"Reverse Path Forwarding of Broadcast Packets," Y. K. Dalal and R. M. Metcalf, *Communications of the ACM*, vol. 21, pp. 1040–1048, Dec. 1978.

Primary Examiner—Douglas W. Olms
Assistant Examiner—Wellington Chin
Attorney, Agent, or Firm—Antonelli, Terry, Stout & Kraus

[57] **ABSTRACT**

A routing algorithm for broadcast packets in packet switching networks, utilizing a "flood-and-forward" technique. In such networks, data are often transmitted in grat quantities from a sensor node to all other nodes in the network, or in a subnetwork, over point-to-point links. Existing broadcast routing algorithms, including multidestination addressing, constrained flooding, minimum spanning tree forwarding, and reverse path forwarding, suffer from an excessive use of bandwidth, a poor choice of routes, or a costly need for memory or computing power. In flood-and-forward routing, periodically a data packet is designated as a Scout packet and is transmitted in a constrained flood broadcast transmission. The Scout packet is identified by a Source Id and a Scout Label. Each receiving node sends a Ack Scout packet to the node from which it first receives a particular Scout packet, acknowledging receipt of that packet. Each relaying node keeps a log of nodes from which it has received Ack Scout packets and sends subsequent, non-scout packets to those same nodes. This flood-and-forward broadcast routing algorithm thus offers the best selection of routes, as in constrained flooding, and the least consumption of bandwidth, as in minimum spanning tree forwarding, while keeping the overhead cost of storage and processing to a low level. With the support of a reliable link service, the algorithm performs well in delivering critical data to all reachable destinations despite to-be-expected losses of packets, links, or nodes.

3 Claims, 10 Drawing Sheets



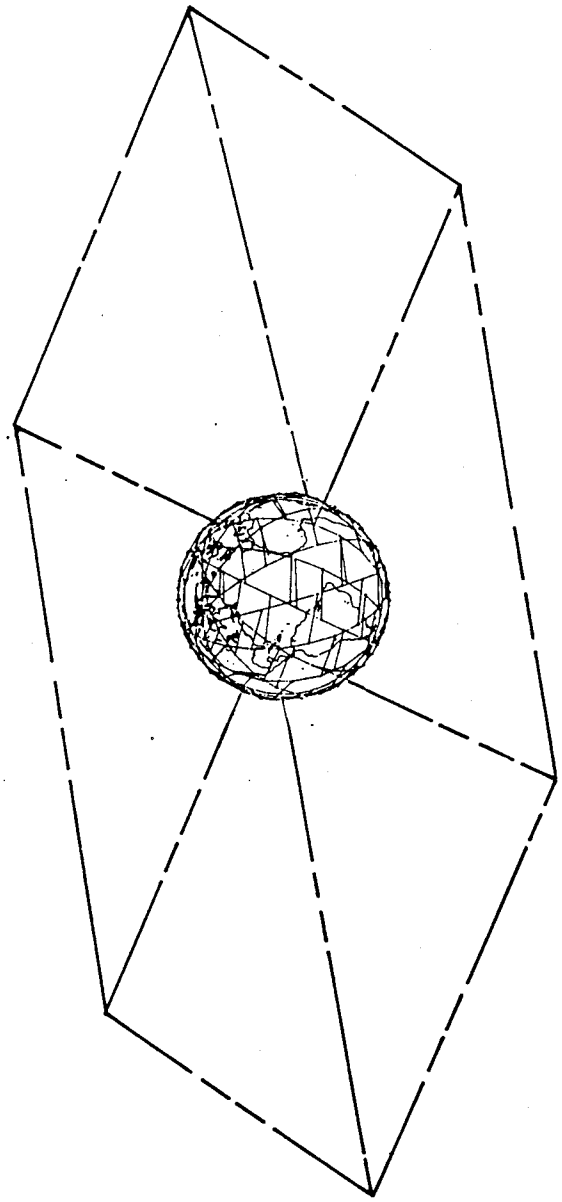


FIG. 1

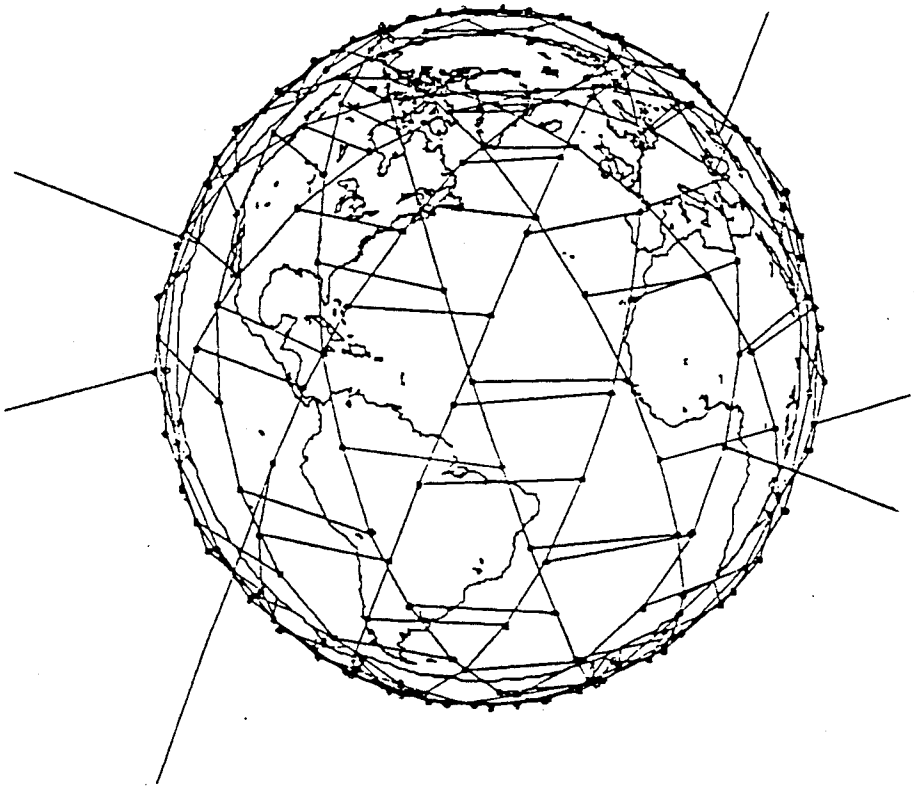


FIG. 2

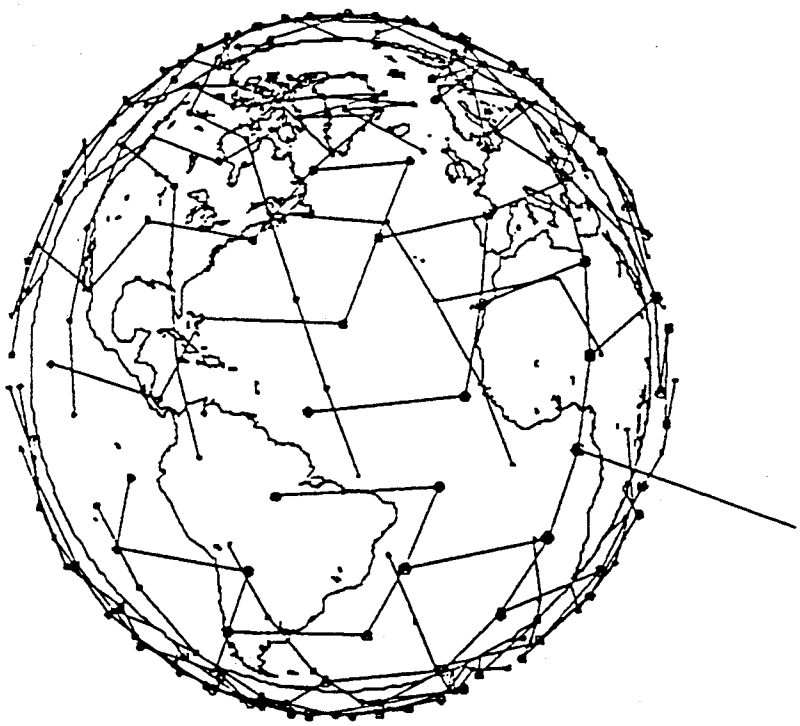


FIG. 3A

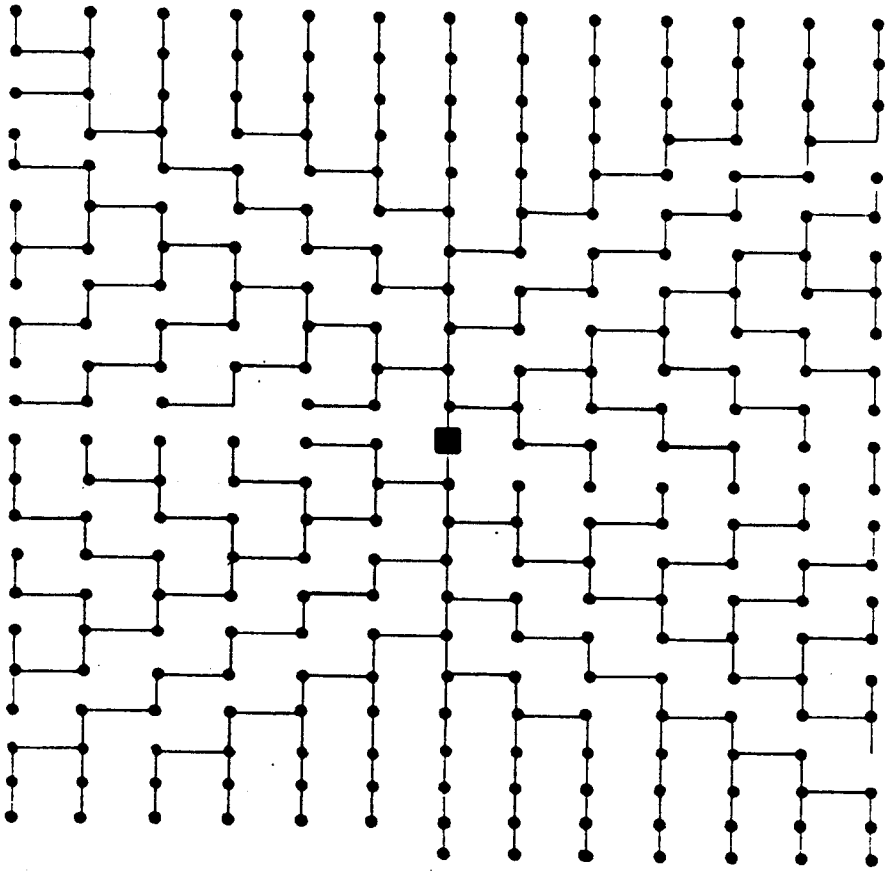


FIG. 3B

```
PROCEDURE GENERATE_BROADCAST(DATA_UNIT) IS
BEGIN
   IF (CURRENT_TIME > SCOUT_LAST_SENT_TIME + NON_FLOOD_PERIOD) THEN
      -- IT'S TIME TO SEND A SCOUT PACKET
      GENERATE_FLOOD_BROADCAST(SCOUT_LABEL,DATA_UNIT);
      SCOUT_LAST_SENT_TIME := CURRENT_TIME;
      INCREMENT_SCOUT_LABEL;
   ELSE IF (CURRENT_TIME > ROUTES_LAST_UPDATED_TIME + ROUTES_LIFE) THEN
      -- ROUTES ARE NOT UP TO DATE
      PUT_PACKETS_ON_HOLD(DATA_UNIT);
   ELSE
      -- USE BROADCAST ROUTING TABLES
      GENERATE_NON_FLOOD_BROADCAST(CURRENT_ROUTES,DATA_UNIT);
   END IF;
END GENERATE_BROADCAST;
```

FIG. 4

```
PROCEDURE PROPAGATE_FLOOD_BROADCAST(SCOUT_PACKET,LINK_ARRIVED_ON) IS
BEGIN
   NOT_YET_SEEN := CHECK_CONSTRAINT_TABLE(SCOUT_PACKET);
   IF (NOT_YET_SEEN) THEN
      ACCEPT_AND_LOG_PACKET(SCOUT_PACKET);
      -- FORWARD SCOUT PACKET
      FORWARD_LINKS := ALL_LINKS - LINK_ARRIVED_ON;
      FORWARD_PACKET(SCOUT_PACKET,FORWARD_LINKS);
      -- SET UP MECHANISM FOR EXTRACTING ROUTES FROM SCOUT PACKET
      SOURCE_ID := SCOUT_PACKET.SOURCE_ID;
      SCOUT_LABEL := SCOUT_PACKET.SCOUT_LABEL;
      ACK_SCOUT_TIMER(SOURCE_ID,SCOUT_LABEL) := CURRENT_TIME +
         ACK_SCOUT_PERIOD;
      BROADCAST_ROUTING_TABLE(SOURCE_ID,SCOUT_LABEL).SEND_TO := NULL;
      BROADCAST_ROUTING_TABLE(SOURCE_ID,SCOUT_LABEL).RECEIVED_FROM :=
         LINK_ARRIVED_ON;
      -- SEND ACK SCOUT PACKET
      PREPARE_ACK_SCOUT_PACKET(SOURCE_ID,SCOUT_LABEL,ACK_SCOUT_PACKET);
      FORWARD_LINKS := LINK_ARRIVED_ON;
      FORWARD_PACKET(ACK_SCOUT_PACKET,FORWARD_LINKS);
   END IF;
END PROPAGATE_FLOOD_BROADCAST;
```

FIG. 5

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.