Bolt Beranek and Newman Inc.

LEVEL III

Report No. 4337

Command and Control Related Computer Technology: Packet Radio

Quarterly Progress Report No. 17
1 December 1978 to 28 February 1979

February 1980

Prepared for:
Defense Advanced Research Projects Agency

BBN Report No. 4337 February 1980

COMMAND AND CONTROL RELATED COMPUTER TECHNOLOGY:

Packet Radio

Quarterly Progress Report No. 17

1 December 1978 to 28 February 1979

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the United States Government.

This research was supported by the Defense Advanced Research Projects Agency under ARPA Order No. 2935 Contract No. MDA903-75-C-0180.

Distribution of this document is unlimited. It may be released to the Clearinghouse Department of Commerce for sale to the general public.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: BBN Report No. 4337	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): COMMAND AND CONTROL RELATED COMPUTER TECHNOLOGY: PACKET RADIO	5. TYPE OF REPORT & PERIOD COVERED: Quarterly Progress Report 1 Dec78-28 Feb 79
	6. PERFORMING ORG. REPORT NUMBER: no. 17
7. AUTHOR(s): M. Beeler, R. Perlman, W.W. Plummer, V. Strazisar, J. Westcott	8. CONTRACT OR GRANT NUMBER(s): MDA903-75-C-0180; ARPA Order-2935
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Bolt Beranek and Newman, Inc., 50 Moulton Street, Cambridge, Massachusetts 02138	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Advanced Research Projects Agency, Information Processing Techniques Office, 1400 Wilson Blvd., Arlington, VA 22209	12. REPORT DATE: February 1980
	13. NUMBER OF PAGES: i + 23
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution of this document is unlimited. It may be released to the Clearinghouse Department of Commerce for sale to the general public.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This research was supported by the Defense Advanced Research Projects Agency under ARPA Order No. 2935.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

packet radio, computer communications, PDP-11, TCP, gateway

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This document describes progress on development of a packet radio network. Activities reported include work on Station Software and Internetworking Research and Development.

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

1. INTRODUCTION

An important component of the Packet Radio project is the station software, providing a variety of control, coordination and monitoring functions. BBN's role in developing this software is to specify, design, implement and deliver programs which perform these functions.

This quarter saw mostly consolidation efforts and preparation for release of new capabilities at a future date. Section 2 covers a number of design and operational issues on which we negotiated with other contractors this quarter. Of special interest are PRTN 265, on congestion control in the Packet Radio network (see section 2.2), and periodic route erasure by PRs (section 2.3.1) instead of the previous station-based route refreshing.

Section 3.1 covers maintenance and minor improvements in the ELF operating system, the BCPL library, and the connection process. Other portions of section 3 describe similar small- to moderate-scale improvements in other station software areas. The two exceptions to this are significant progress in preparing the CAP5 Labeler (section 3.1.2.1), and the PR down line load process (section 3.1.3), the latter of which is nearly ready for delivery.

Section 4, dealing with Internetworking issues, presents the release of TCP version 2.5.2 and the readying of the new version 4 TCP. Also in section 4 is a discussion of gateways with an improved routing algorithm, which are now in operation.

Section 5, on hardware, includes discussions of Error Control Unit testing and the installation of operational Improved Packet Radio units in the BBN testbed.

2. MEETINGS, TRIPS, PUBLICATIONS

2.1. Meetings and Trips

BBN personnel attended a TCP meeting at the ARPA office in Washington, Monday, December 4. The meeting was followed by a visit to DCEC in Reston, Virginia see the Electronic Data Network and for initial compatibility testing of TCP4 implementations. At that point in time BBN was not ready to demonstrate TCP4, but did supply a "testing gateway" for use by participants. The testing gateway simulates a very bad communications network by dropping, duplicating and corrupting packets and is used to evaluate how resilient host TCP4/IP4 are.

BBN hosted visits from Bob Braden of UCLA on December 1, 1978 and Aage Stensby of NDRE on December 6, 1978. In both cases plans were made for TCP compatibility testing. BBN promised to write a user-mode support program to assist NDRE. This program was to listen for connections from remote TCPs on two sockets. One socket would simply copy (TELNET character) data from the connection into a disk file, while the other would copy the contents of that file to the network. This program will be implemented on an available time basis and design has not started.

BBN personnel attended a Packet Radio Working Group meeting at SRI, followed immediately by an Internet Group meeting at ISI. At both sites new TCPs were installed on their TENEX machines. These installations were complicated by the existence of local modifications to the monitor which had to be fixed up so that it was consistent with the (96-bit) NCP being installed.

ISI also hosted the "TCP Bakeoff" during which the various TCP implementers tested their systems against each other. Aside from the usual problems with checksums, the results were strikingly good. The BBN TENEX TCP 4.0.0 survived all tests.

At the PRWG meeting this quarter, held January 22-24 at SRI, BBN personnel participated in or led several discussions besides those regarding Internet and TCP issues described above. These other issues included periodic route erasure (see section 2.3.1), PR down line loading (section 3.1.3), multiple Alter Memory (AM) and Display Memory (DM) XRAY command packets (section 3.1.4) not being serviced properly by PRs, a command packet to force the target PR to down load, and plans for further ECU testing (section 5.1). We also suggested that rudimentary multistation capability (PRTN 260, see QPR 16; and PRTN 264, see section 2.2) can be included in the initial release of Channel Access Protocol version 5 (CAP5), although this was judged too large a step by the group.

The PRWG meeting was also the opportunity we used to present some new thoughts on congestion, and in particular the cycling of the transmit queue and the usefulness of "channel busy" as an indicator of radio channel congestion level. These ideas were set out in PRTN 265 (see section 2.2).

2.2. Publications

PRTN 264, "Changes Necessary for Rudimentary Multistation Capability"

This PRTN itemizes the software changes required to implement the rudimentary multistation capability described in PRTN 260. Since only a few, rather straightforward changes are necessary, this PRTN focuses attention on the feasibility of implementing rudimentary multistation relatively soon. Five station changes are needed, two in TIUs, and ten in PRs.

PRTN 265, "Issues in Congestion Control: Detection and Current Routing Design"

This paper discusses the measurable quantities which could lead to the detection of congestion, and the control variables which might be used to control congestion. The factors involved are related to the current routing mechanisms in the network. One important conclusion is that the FIFO transmission scheme used for packet radio transmissions is inefficient. The suggested alternative is a cyclic transmission scheme in which more then one packet may be simultaneously unacknowledged if they are intended for different, neighboring packet radios. This result was verified in a simulation study by UCLA which is described in PRTN 273. Another conclusion of this paper in the primacy of "channel busy" as an indicator of congestion.

2.3. Negotiations and Informal Documents

2.3.1. Periodic route erasure

We seem to have partially redesigned PR route-handling in the middle of the PRWG meeting. In talking about the new approach afterwards at BBN we found an unfortunate impact on CAP5 routing. If our understanding of the current approach is correct, we believe that a small change to PR route erasure (as described below) is very important.

Decision at PRWG Meeting

- PR only erases a route if it needs the slot for a new route.
- PR keeps track of route usage by a time stamp and will not erase the route until its lack of use goes over a certain threshold.
- PR will erase the route which has been used least recently if a slot is required.

Problem

If the TIU is only interested in sending packets to a few destinations some routes may rarely, if ever, get garbage collected. Since the CAP5 station will try to reassign routes traveling over a bad link, rather then simply erasing them, a significant portion of station resources may be tied up keeping track of and updating old, unused routes.

Suggested approach

We suggest that two thresholds be used for decisions about route discarding. The first, and current, is the minimum time before the PR can optionally garbage collect an unused route slot for a new route. Its purpose is to provide slots for new traffic so the traffic won't be forced to funnel through the station which is both slow and costly.

The second threshold is the maximum time an unused route should be allowed to remain in the PR's route table. After this threshold is reached route erasure is mandatory regardless of need for reuse of the slot. The purpose of the second threshold is to reduce control traffic and route computation by the station. The mandatory route erasure would be a much longer time interval than the optional route erasure. Perhaps on the order of minutes for optional, and on the order of hours for mandatory.

Why it works

The station won't realize the route has been erased because the mandatory erasure is based on time since last use rather than time since creation, and will retain knowledge of the route until there is a problem on one of the links. At that time it may choose to send out a new route. It will then be informed that the route is absent from the PR and will erase the route from the station's own table.

This approach limits the useless control traffic to one exchange. If the route hadn't been removed, the station would continue to compute new routes for the PR.

Resolution

The eventual resolution of this problem is expected early next quarter.

2.3.2. Route computation algorithm for the Labeler

We plan to do the route computation using Floyd's Algorithm for Shortest Path (ref: R. W. Floyd, "Algorithm 97, Shortest Path," CACM 5(6), June 1962).

```
  DO 2 I=1,NN                     ;init all node pairs
    DO 2 J=1,NN
      IF(C(I,J)) GO TO 1
      D(I,J)=MAX                  ;nodes not adjacent
      RM(I,J)=0
      GO TO 2
1     D(I,J)= DD(I,J)             ;nodes are adjacent
      RM(I,J)=J
2     CONTINUE
  NN1=NN-1
  DO 5 K=1,NN1                    ;K is intermediate node
    DO 4 I=1,N                    ;I=source, J=destination
      DO 3 J=I,N                  ;node J starts at I
      IF(D(I,K)+D(K,J).GE.D(I,J)) GO TO 3
      D(I,J)=D(I,K)+D(K,J)        ;a new shortest path
      RM(I,J)=RM(I,K)             ;update route
3     CONTINUE
4   CONTINUE
5 CONTINUE
```

DD will contain the link qualities and the transfer between DD and D will contain the intelligence. Since routes need to have good link Q's in both directions, D(I<,J) will have the lesser of DD(I,J) and DD(J,I). Therefore D and RM (routing matrix) will be mirror-image matrices (helping to alleviate the station's memory crunch).

Initially, we expect all modifications to the desirability of a hop to be reflected in lowering or raising the apparent link Q in the D matrix.

Information arriving from PDPs will be stored in the DD matrix until a new route request or route failure PDP or DLROP is received. At that time RM will be recomputed and the appropriate action taken.

These design data were communicated to PR contractors this quarter to aid them in understanding network operation and in planning their activities. In particular, UCLA personnel wished to incorporate these algorithms into their PRN simulator.

2.3.3. Miscellaneous negotiations

Late this quarter we corresponded with Collins about cycling the transmit queue in PRs. We had published PRTN 265 (see section 2.2), which discusses the usefulness of this action in reducing congestion. We are concerned that the algorithm in the PRs provide a full cycling through all queued packets, rather than, say, merely allowing the first two packets (instead of only the first one) to block the rest, or biasing transmission so strongly in favor of earlier packets that only rarely are any but the first ever in fact transmitted. Either of these partial implementations would severely compromise the benefits of cycling the transmit queue, thus still encouraging congestion. The resolution of the actual implementation was not clear at the close of this quarter.

During this quarter we negotiated with Collins the final details of PR down-line loading. Most of the important questions concerned the format of the "object" file Collins supplies, which contains the code to be loaded into the PR. Our system of transferring this code to the station disk involves use of a reformatter program which must parse the object file. Thus we needed to know just which options of the National Semiconductor IMP-16 assembly language are used by the code. In particular, relocatable code is never generated. The reformatter checks for such conditions and prints warning messages when they occur. Problems with incorrect object files are thus often caught at this early stage, before transfer to station disk, loading, or running in PRs.

Sluggish behavior of internet and intranet (when traversing the station) traffic was reported by SRI this quarter. Its cause was traced to their practice of leaving the station's packet printer enabled all the time. As had been discussed before, this can become a bottleneck if station traffic volume is high. We suggested enabling the packet printer only occasionally, for checking or to diagnose peculiarities.

The PR attached to the gateway will have routes assigned to several destination IDs. We noticed the possibility that the route table in this PR might become full, causing overflow traffic to be forwarded by the station. This in turn would cause increased delay, possible congestion, and attempts to assign routes for the overflow traffic. When honored by the gateway's PR, these route assignments would displace other routes which may still be in use. This would cause a continuing shuffle of which traffic had routes, or a "thrashing" effect in the routing. We suggested that the route table in any critical PRs, particularly the gateway's PR, be enlarged to avoid these effects.

During this quarter SRI has identified what they believe to be evidence that the cache on the Fort Bragg station PDP-11 is not "transparent", that is, that modifications to the ELF operating system are necessary for use of the cache. We are investigating their concerns and expect to resolve the issue in the coming quarter.

The Fort Bragg PRN site is expected to have only one ARPANET IMP port available to it for the near future. This presents a problem, since user traffic should avoid the station whenever possible, to reduce delays and prevent congestion. So both the station and a gateway need ARPANET ports. ARPA and SRI made a suggestion to solve this problem, as follows. Use an LSI-11 Port Expander (PE), modified to expand a PRN port instead of an ARPANET port. Also integrate into it a standard, 2-port minigateway (mg). The LSI-11 mg/PE has three port connectors; one goes to a PR, one "expanded" PR port goes to the station, and the other expanded port goes to the mg, whose connector goes to the ARPANET. Access between the station and the ARPANET would be through the PE. This would use only one PR of those at Fort Bragg, which are a scarce resource during initial deployment there. We made a two-step modification to this plan. First, put an additional port on the mg, which will provide the station's "ARPANET" connection. (ARPANET in quotes, since this link would actually constitute a network itself.) Second, use another PR attached directly to the 3-port mg, where the PE attached, eliminating the PE. This costs one more PR, but we feel the chances of making this work soon are much better than with the ARPA-SRI scheme. In particular, reliable PE operation has not yet been demonstrated; and the mg/PE integration task is of uncertain difficulty.

We sent ARPA a description of mini-gateway and port expander hardware needed at BBN. During the coming year, BBN will be developing mini-gateways and integrating these with port expanders for use as ARPANET/SATNET gateways and ARPANET/Packet Radio Network gateways. In order to develop and maintain these systems, we requested copies of the hardware that will be used in the Ft. Bragg mini-gateway and port expander and in the SATNET mini-gateway and port expanders. We also requested the hardware to support a mini-gateway and port expander between the BBN Research Computing Center Net (RCC Net) and the ARPANET. This configuration would allow us to test the mini-gateway and port expander in a more operational environment as the gateway between the RCC Net and the ARPANET is in constant use to access machines on the RCC Net. It would also allow us to test more complex internet configurations and would better exercise the alternate routing capabilities of the gateways as this gateway would provide a second path (parallel to the PTIP gateway) between the RCC Net and the ARPANET.

SRI reported problems with TCP and/or the gateway. In one case we investigated, the SRI station's IMP11-A hardware had failed. In a second case, we assisted SRI in tracking down the cause of BBNC intermittently refusing TCP connections; the cause was an extremely high load average causing the TCP server to be unable to do a "listen" after a prior "listen" had been matched up to a connection.

3. THE PACKET RADIO NETWORK

3.1. Station Programming and Testing

3.1.1. ELF, library and connection process

We continued to support software used in the Packet Radio Network station. We made several corrections to the ELF software. Since converting the ELF XNCP to use 96-bit leaders, we discovered that the IMP may send 32-bit leaders if the interface is reset. To avoid this problem, we modified the ELF to send 96-bit NOPs whenever it received a message with a 32-bit leader. We also corrected a problem in the ELF Halt I/O routine. This routine did not properly halt I/O on interprocess ports. We believe that this error was present since the ELF was written and that it may have been responsible for past anomalies in the Packet Radio Net station.

We also corrected a problem in the BCPL library timing routines that are used by several station processes. In addition, the BCPL library now translates the BCPL function "finish" into an ELF call "freeze", instead of a halt. This permits easier debugging of programs which terminate unexpectedly.

We modified the connection process to allow the gateway in the Packet Radio Net station to send TOPs. The new gateway software sends messages to itself over the Packet Radio Net interface in order to determine the status of this interface. In order for the labeller and connection process to route these packets to the gateway, it is necessary for the gateway to send TOPs to identify itself as a device on the station PR.

3.1.2. Labeler

3.1.2.1. CAP5 development

We have begun work on the required changes for the CAP5 Labeler. The first phase, a redesign of the connection handling, is possible to debug while still under the CAP4.9 protocol. The new design must allow the packet radio to open an SPP connection to the Labeler. This requires a new type of connection to be created called the listening connection. Previously, the packet radio could not initiate the opening of an SPP connection.

The listening connection is set up between the connection process and the Labeler to allow any PR, who doesn't already have an SPP connection, to attach to the listening connection. At the time of attachment, the connection is tamed and becomes a normal SPP connection.

Since the listening connection brought the connection types handled by the Labeler to three (wild, tame SPP, listening SPP), it was felt that one general set of routines should be built to handle all cases. Therefore, the specialized routines in the Labeler which handled only wild (non-SPP) or normal SPP connections were replaced. To make this transition easier and to save memory in the station, the BCPL structures containing the connection handling information were redesigned as well.

Thus we were able to design, build and test a significant portion of the new CAP5 Labeler while still running on the CAP4.9 network. There was great value in debugging the new connection handling procedures while using throughly tested packet radio software in the CAP4.9 network. We were also able to test the new listening connection by causing XRAY to initiate SPP connections to the Labeler.

3.1.2.2. CAP4.9 developments

Unfortunately, we also had to do further development in the old, CAP4.9 Labeler at the same time. This forced the overhead of maintaining two complete sets of Labeler code and improving both sets as development proceeded.

The CAP4.9 Labeler was modified to support down line loading of PRs. It notified the loader process when a ROP containing a load request was received. The information passed included the PR requesting load, the PR offering to load, the type of PR to be loaded, and how far the loading had proceeded. To make room for this code, the number of PRs the Labeler could support was decreased from 31 to 30.

The CAP4.9 Labeler was further modified to desist from periodic erasure of PTP routes, in accord with negotiations at the PRWG meeting this quarter. The PR now has responsibility for ensuring that unused routed are discarded.

3.1.3. PR down line load process

During this quarter the PRLOAD process, to down line load Packet Radio units over the radio channel, was completed and shown to work in the BBN two-PR radio net. A small delay resulted from a confusion in documentation from Collins; the initial description of the down load command packet had only two words of text header, one to identify the packet as a down load command packet and specify a sequence number in its right half, and one to specify the ID of the PR being loaded. The later, corrected documentation showed a third word to hold the load packet sequence number. Fortunately, conversion to the new format was not a major difficulty, although determining that the format change was the cause of load malfunctions took a noticeable effort.

The latest CAP protocol software, version 4.8, fits in 30 load data command packets. When loading the non-station PR at BBN, SPP protocol permits positive sequencing, and a full load takes about 4 seconds. When loading the station PR, SPP cannot be employed since there is no loader PR to which the SPP connection could be established. The present implementation uses a small explicit delay between packets to avoid overrunning the raw (non-SPP) connection. Consequently, loading the station PR takes longer, about 9 seconds.

Documentation of PRLOAD and its support programs, REFOR (PR code reformatter) and DLLDMP (which writes, or "dumps", down line load data onto station disk), is in progress. Delivery of the PRLOAD system early next quarter is anticipated, after completion of documentation and of testing in the SRI net.

3.1.4. XRAY cross-radio debugger

During this quarter modifications were made to XRAY, the station module which provides debugging access from the station operator's terminal to PR units in the net. These modifications are in support of the use of Improved Packet Radio units (IPRs). Since IPR memory occupies 20 bits of address space, instead of 16 bits as in the older Experimental PRs (EPRs), XRAY has been modified to manipulate PR addresses as double word quantities, including terminal input and output.

Secondly, this time has been chosen as a convenient one for PRs and the station to switch over from control packets containing executable code, to command packets containing only parameter values. Use of executable code provided flexibility and reduced memory requirements during earlier stages of PR net development. The instruction sets of EPRs and IPRs differ, however; so different executable packets would be necessary depending on what type of PR was the target. This makes the conversion to command packets appropriate at this time.

Debugging of this modified XRAY has begun in cooperation with SRI personnel in the SRI net. A minor problem in packet format specification has been found and fixed. Re-installation of IPRs during January permitted further XRAY debugging at BBN, but personnel were also needed for ECU testing and PRLOAD development, with the result that delivery of the modified XRAY is postponed to early in the next quarter.

3.2. Support

XNET has been modified to talk to logical hosts through port expanders. This will permit continued use of XNET to load, debug and correspond with stations and gateways placed on port expanders. At the close of this quarter, there had not yet been an available such host with which to test this capability.

3.3. CAP4.8-compatible Station

With cooperation from SRI personnel, we tested the CAP4.8 measurement process and connection process in the SRI net. We then delivered these processes to both SRI station disks. The only change in the measurement process was to use the new command packet to initiate measurements, rather than an executable control packet. The change in the connection process was to make use of the new route suppression bit in packet headers to determine which forwarded packets to tell the Labeler about, for potential point-to-point route assignment. The design of this new bit was discussed in QPR 16.

We also delivered a new version of PRDATA, which prints CAP4.8-format measurement packets from PRs.

Although CAP4.8 responded properly to the new measurement command packet and sent correctly formatted measurement packets to the station, some of the statistics reported were incorrect. We reported the details to Collins and provided them with a PRDATA printout of the bad data to aid in debugging.

4. INTERNETWORKING

4.1. Transmission Control Program (TCP)

TCP 2.5.2 sources were released to ISI and SRI for installation on their TENEX machines. As previously mentioned (section 2.1), the installation at both sites was complicated by the local modifications which had been made to the TENEX monitor sources. At SRI it was only the code for (their) GTBLT JSYS which was affected and rewriting the code made it all work. ISI, however, has made extensive changes to the sources, mostly having to do with trying to have one set of sources from which a TENEX monitor can be built for either a KA10, KI10, or KL20 processor. Integrating new code into such a highly modified set of sources was more difficult than expected and had not been completed by the end of the meeting.

This same version of TCP (2.5.2) was modified in a fairly direct manner to obtain the first version 4 TCP (4.0.0). Although some protocol features were not supported, the packet format was changed and much of the internal work done. This version was installed on BBNB and was used for the "TCP Bakeoff". During the Internet meeting and Bakeoff, the existence of a bug was revealed. This bug was cured shortly after the meeting.

A suggestion has been made which will provide a dramatic increase in TCP performance. Basically, it is a way around the need to map pages of the user space into the monitor space in order to transfer data. The hardware pager is to be used instead. Since the TCP (part of the Internet process) never leaves monitor mode, that process's user mode space can be switched to be identical with the caller's space. References to the user's buffers will then be done by the standard "move to/from previous address space" instructions, even though the caller is in a totally different job than the TCP. The first implementation using this idea will be tried out soon using the DEC 2020 which has been made available for TCP testing.

4.2. Gateways

During this quarter, we delivered new versions of the software for both the ARPANET/SATNET gateways and the ARPANET/Packet Radio Net gateways. These versions of the gateways are the first to use gateway alternate routing as specified in IEN #30 and presented at the London meetings in May, 1978. Earlier versions of the gateways forwarded internet traffic using routing tables assembled into the gateways. These versions of the gateways provide significantly better service in the catenet by dynamically routing internet traffic around failed gateways and networks.

Briefly, the gateway alternate routing algorithm operates as follows. Each gateway determines the status (up or down) of its network interfaces and the status of its neighbor gateways. The neighbors of a gateway, A, are those gateways which are connected to the same networks as gateway A. Using this information, the gateway computes its distance (in number of networks) to each network in the catenet, and sends this distance vector to each of its neighbor gateways. The gateways forward internet traffic on the shortest path (measured in number of networks traversed) to the destination network. If there are several shortest paths to a network, the gateway load splits traffic for that network over all the shortest length paths.

One problem often encountered in such a routing scheme is that traffic can loop through some set of gateways. In order to prevent looping in most cases, the algorithm has been modified as follows. A gateway sends its actual distance to a network to its neighbor gateway only if it is no further from the network than its neighbor. Thus a gateway can only send traffic for a network to gateways that are closer to the network than itself. For example, in the catenet diagrammed below, A, B, C, D, and E are gateways and N is a network. Because gateway A is further from network N than gateway B, gateway A will report to gateway B that it is infinitely far from network N. Thus, gateway B will forward its traffic to N through gateway C. However, if the network between gateway B and gateway C fails, gateway B will report to A that it is infinitely far from network N. On receiving this information, gateway A will begin forwarding traffic for network N through gateway D. Gateway A will also report its correct distance to network N to gateway B, and gateway B can then begin forwarding traffic for network N through gateway A. (This algorithm is explained in more theoretical detail in section 4.3.)

```
A ----- B ----- C ---- N
|               |
|               |
|               |
|               |
D ------------- E
```

Gateways exchange messages with each of their neighbors periodically in order to determine their connectivity to their neighbors. However, routing updates are only sent in response to a change in the status of the gateways or networks, i.e., a gateway or network fails or recovers, or a gateway interface to a network fails or recovers. By sending routing updates only in response to changes in the catenet, the overhead of exchanging routing information is reduced.

These gateways are also able to use gateways, such as the BBN Pluribus TIP, that do not implement this algorithm. A distance vector, containing distances from the gateway to each network, is calculated for each gateway that does not participate in this routing algorithm. This distance vector is assembled into all gateways that participate in routing. If there is no other path to a network, the gateways will use these static distance vectors to compute a shortest path to the network using the non-routing gateways.

By the end of the quarter, the new versions of the gateways had been in operation for nearly a month. Although in the present catenet configuration, gateways usually forward traffic only for networks to which they are directly attached, we did observe several cases in which the gateways did choose alternate routes to avoid failures in gateway interfaces or networks. For example, if an ARPANET/SATNET gateway, which sends continuous monitoring reports to the Gateway Monitoring and Control Center on BBNE on the ARPANET, becomes disconnected from the ARPANET, it will send traffic for the ARPANET, including the monitoring reports, via the SATNET.

4.3. Improvement to ARPANET-style Routing

Because the ARPANET routing algorithm is more efficient, in terms of amount of computation required by the nodes and traffic overhead, than either of the algorithms proposed in PRTNs 241 and 242 (see QPR 13) for gateway routing, we decided to adapt the ARPANET routing to meet our needs. The most important modification we made to the ARPANET routing algorithm was to fix the following problem:

```
A--------B--------C
```

In this example, B reports that it is 1 hop away from C. A therefore reports that A is 2 hops from C. If the link from B to C breaks, B will decide, not that C is unreachable, but that B is now 3 hops from C, because B has a neighbor, A, which is 2 hops from C. A and B will continue to loop traffic for C through each other until they count up to the maximum legal value for distance from C.

We fixed this problem in the following way. When gateway G reports G's distance to destination DEST, dist(G,DEST), to neighbor gateway N, G reports:

dist(G,DEST) if dist(N,DEST) .ge. dist(G,DEST)

and reports

infinity if dist(N,DEST) < dist(G,DEST).

This scheme (reporting infinite distance to downstream neighbors) helps prevent loops because nodes will not have access to information about paths through themselves. If the route to a destination suddenly gets longer, a gateway will temporarily think the destination is unreachable and report infinite distance. If a neighbor has a path to the destination that does not involve the reporting gateway, it will immediately report the true distance to the gateway. If the neighbor's best route did involve that gateway, the report of infinity from that gateway will cause the neighbor to immediately update its table to infinity, or a route not including that gateway.

There are still cases where temporary large loops can form, as in a ring of gateways suddenly cut off from some node not in the ring. Distance information to the node can circle around the ring, constantly incrementing. We feel this is not a serious problem. First of all, such connectivity does not occur in the (present) catenet. Second of all, large loops are not as serious as small ones. Third of all, in such a large loop, it would only take a few circles through the loop for the distance information to count up to infinity.

5. HARDWARE

5.1. ECU Testing

The Error Control Units (ECUs) built by the firm ACC are used to provide a reliable link with characteristics roughly those of a local or distant host ARPANET "1822" connection. Use of these is planned for Fort Bragg, where the station must be connected to a distant IMP whose remoteness prohibits use of ordinary local or distant host cable. Some reports of incompatibilities between ECUs and station software arose recently, and during this quarter we have pursued testing activities to locate any true incompatibilities. These activities are being pursued at the direct request of ARPA, to support station development and installability.

Several potential problem areas were identified, as follows.

(1) The station's IMP11-A interface driver may not follow the letter of Report 1822, in terms of having an explicit delay after asserting Host Master Ready and before enabling input or output. Listings were sent to ACC. In any case, lack of the delay should only garble a message or two, not hang the interface.

(2) The ELF XNCP had a bug which could cause it to fail to initialize the IMP to operate in 96-bit leader mode. Only packets with 96-bit leaders will be processed by the station. This bug was corrected.

(3) ACC supplied two ECUs on-site at BBN. Using these, we found an error in (Pluribus) IMP documentation, resulting in problems in ready line sensing. This error was corrected.

(4) The standard IMP11-A diagnostic declares the IMP down if the ready line does not come up in one second. Handshaking between the ECUs over their 9.6 KB line delays this by 1.5

seconds though, so the diagnostic failed. It has been patched.

(5) The ECUs sent by ACC were local host but our installation presents a 200 foot distance. This probably results in signal skew, which explains picking the first bit in various words, which was seen. ACC is shipping parts to convert the ECUs to distant host. As this quarter closes, we are awaiting these parts and a visit from ACC personnel as arranged by ARPA, to aid the ECU hardware debugging efforts.

5.2. Packet Radio Units

Following the burn-out of the new IPRs last quarter (see QPR 16), we coordinated with Collins the installation of a new pair at BBN this quarter. Collins personnel were in attendance to ensure proper installation, repair or replace any faulty components, and test/align the Radio Frequency (RF) heads. Reflective material on the roof of our building prevented connectivity between the PR on the first floor and that on the seventh, whose antenna is on a mast on the elevator hut on the roof. Consequently, both PRs have been installed on the first floor and connected by coaxial cable (and appropriate attenuators).

Both units are now operating properly. With actual RF radiation now possible, concerns over microwave safety arose. We have made inquiries of industrial safety hygienicists in this regard, and forwarded their response to ARPA and Collins. We anticipate further discussion of this issue during the following quarter.

Full operability of IPRs at BBN is a major milestone, permitting progress in debugging and testing station software to move rapidly.

5.3. Miscellaneous Hardware Work

This quarter was affected by two memory problems. Station PDP-11 number 2 suffered failure of one memory board, which has been replaced. BBNA, the service host on which most of our development work is carried out, was plagued by memory parity problems. It is hoped that these are fixed, so progress will resume at a normal pace.