Bolt Beranek and Newman Inc.

LEVEL III

Report No. 3645

A108325

AD A108521

Command and Control Related Computer Technology
Part I. Packet Radio
Part II. Speech Compression and Evaluation

DTIC ELECTE DEC 1 0 1981 D H

September 1977

Submitted to:
Defense Advanced Research Project Agency

BBN Report No. 3645 September 1977

COMMAND AND CONTROL RELATED COMPUTER TECHNOLOGY

Part I. Packet Radio
Part II. Speech Compression and Evaluation

Quarterly Progress Report No. 11

1 June 1977 to 31 August 1977

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.~~ 3645	2. GOVT ACCESSION NO. AD-A108 521	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) COMMAND AND CONTROL RELATED COMPUTER TECHNOLOGY, Part I, Part II.		5. TYPE OF REPORT & PERIOD COVERED Quarterly Progress Report 1 June77-31Aug77
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) M. Beeler, J. Burchfiel, A.W.F.Huggins, J. Makhoul, R.S.Nickerson, R. Perlman, W. W. Plummer, V. Strazisar, J. Sussman, R.S. Tomlinson, R. Viswanathan		8. CONTRACT OR GRANT NUMBER(s) MDA 903-75-C-0180
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 Technology Office 1400 Wilson Boulevard, Arlington,VA 22209		12. REPORT DATE September 1977
		13. NUMBER OF PAGES 48
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, station gateway, cross-radio debugging, speech compression, vocoder, linear prediction speech synthesis, Objective speech-quality evaluation, multidimensional scaling.

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

This document describes progress on (1) development of a packet radio network, (2) speech compression and evaluation. Activities reported under (1) includes work on Station Software and Internetworking Research and Development; under (2) Development of a new source or excitation model for the synthesizer of the narrowband LPC vocoders, with the objective of enhancing the naturalness of the synthesized speech; development and validation of several procedures for objectively evaluating the quality of LPC vocoded speech.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

060100

Bolt Beranek and Newman Inc.

Report No. 3645

Command and Control Related Computer Technology
Part I. Packet Radio

September 1977

Submitted to:
Defense Advanced Research Project Agency

TABLE OF CONTENTS

I. INTRODUCTION

The Packet Radio project relies heavily on station software for a variety of control, coordination and monitoring functions. The role of BBN in developing this software is to specify, design, implement and deliver programs which implement these functions.

During this quarter delivery was made to Stanford Research Institute, in their role as System Evaluation and Technical Direction contractor to ARPA for the Packet Radio project, of a major new collection of station software modules, plus new releases of several modules previously transferred to SRI. While we have released changes as appropriate during the past year, this marks an important milestone in the development of the Packet Radio project. The delivery includes several improvements, which are discussed in the appropriate sections below, as well as documentation and familiarization activities.

A second significant accomplishment this quarter was the arrival and installation of hardware to upgrade the second BBN PDP-11 so that it is fully functional as a station. This work is covered in section V below.

Continued research and development in the area of internetworking is another important aspect of our progress this quarter, and is covered in section IV.

II. MEETINGS, TRIPS, PUBLICATIONS

During this quarter BBN personnel participated in several formal and informal exchanges as a part of our Packet Radio effort. A Packet Satellite Project meeting in July at NDRE was attended by a representative from our group, as was the August Satellite Network meeting at Linkabit in San Diego.

Virginia Strazisar attended the Ninth Packet Satellite Program Working Group Meeting at Kjeller, Norway. At this meeting, the design of communication paths between gateway modules was presented. It was decided that fake hosts residing in the gateway machine and the gateway would have separate logical addresses. The Host/SIMP module will interface to both the fake hosts and the gateway and will multiplex and demultiplex messages from these modules to and from the SIMP. The fake hosts will also be able to communicate with the SIMP through the gateway using Internet messages. Thus, one set of fake hosts can interface to the Satnet either as local (to the Satnet) hosts or as Internet hosts.

The gateway PDP-11 was installed at the Norwegian Defense Research Establishment (NDRE) at the time of the Kjeller meeting. Documentation was prepared to explain the cross-net debugger, the fake hosts, and the fake host/gateway interface. Virginia Strazisar spent three days with personnel at NDRE helping to check out the hardware and to acquaint them with use of the cross-net debugger for loading and running the gateway and fake host software. Although the IMP11-A on the gateway machine worked initially, it failed soon after installation. BBN worked with personnel at NDRE and DEC to help debug this hardware. The failure was eventually traced by DEC to problems in the CPU and DR11Bs (modules in the IMP11-A which interface it to the PDP-11 UNIBUS), and the IMP11-A is now operational.

At the San Diego Packet Satellite meeting, the current status of the satellite net was discussed, including hardware problems and experiments. Also we discussed gateway issues, mainly the problem with lack of memory. The basic design of the host/SIMP protocol module was discussed and various suggestions were made about queuing strategies within the module that would work well with the ordering strategy in the SIMP.

Closer to home, BBN personnel attended a July TCP meeting at MIT. At these meetings, whose interest lies in areas larger than the Packet Radio network itself, we are both representing the needs which the Packet Radio project has for compatibility with these contexts, and contributing to the advancement of functional design in the internetwork arena.

As part of our familiarization activities to train SRI personnel in the use of the new station software delivered to them this quarter, we hosted a visit from Jim Mathis and Jim McClurg of SRI in July. We invited them to spend a day with us in order to become familiar with the new station software, which was almost ready to deliver. We gave them drafts of the new documentation, and demonstrated the new information process, station control process, and measurement process. Part of the demonstration consisted of running the software at SRI, collecting measurements from three PRs, and sending the data with TCP to our receiver/printer at the SRI-KA TENEX. We also discussed TIU measurements, and agreed on how the station measurement process will control TIU cumstats and traffic sources/sinks and collect the data. Parameters and packet contents were specified, as well as the handling of connections. TIU measurements are scheduled for implementation next quarter.

To develop adaptive routing mechanisms for traffic among gateways, we have been considering the work of Gallager of MIT. We met with Professor Gallager during this quarter to clarify some of the issues we had identified while studying his writings. As a result of our better understanding, we are now preparing a proposal for gateway-gateway routing based on his work.

During this quarter we were also engaged in discussions with Collins Radio regarding software development and delivery schedules. Negotiations on this issue promise to make the next round of software checkout, probably to occur in the next quarter, smoother and faster than in the past.

The Packet Radio Station Notebook contains documentation describing the design and operating procedures of the station. Concurrent with the delivery of a major new version of station software this quarter, we have assembled an update package to the

Notebook. This package contains several completely new documents, as well as current revisions of several documents distributed previously.

Packet Radio Temporary Notes issued by BBN this quarter are:

* PRTN 174 - Revision 3, Packet Radio Network Station Labeling Process
  This revision updates the description of labeling to include the handling of DROPs (Distress ROPs). DROP handling by the labeler was described in our last quarterly report.

* PRTN 212 - Revision 1, Specification of Measurement File Entries
  This PRTN defines and describes the content and format of all entries made in the measurement file by the station measurement process. It is complete and correct for the measurement process delivered this quarter.

* PRTN 232, SPP Retransmission Count Field
  To improve the congestion level in the PR net, especially in future configurations with higher traffic levels than at present, PR units should filter out (suppress propagation of) duplicate packets generated within the net. End-to-end retransmissions, however, should not be filtered out. PRTN 232 specifies the change to SPP protocol which BBN and Collins agreed on to make this filtering action possible to implement.

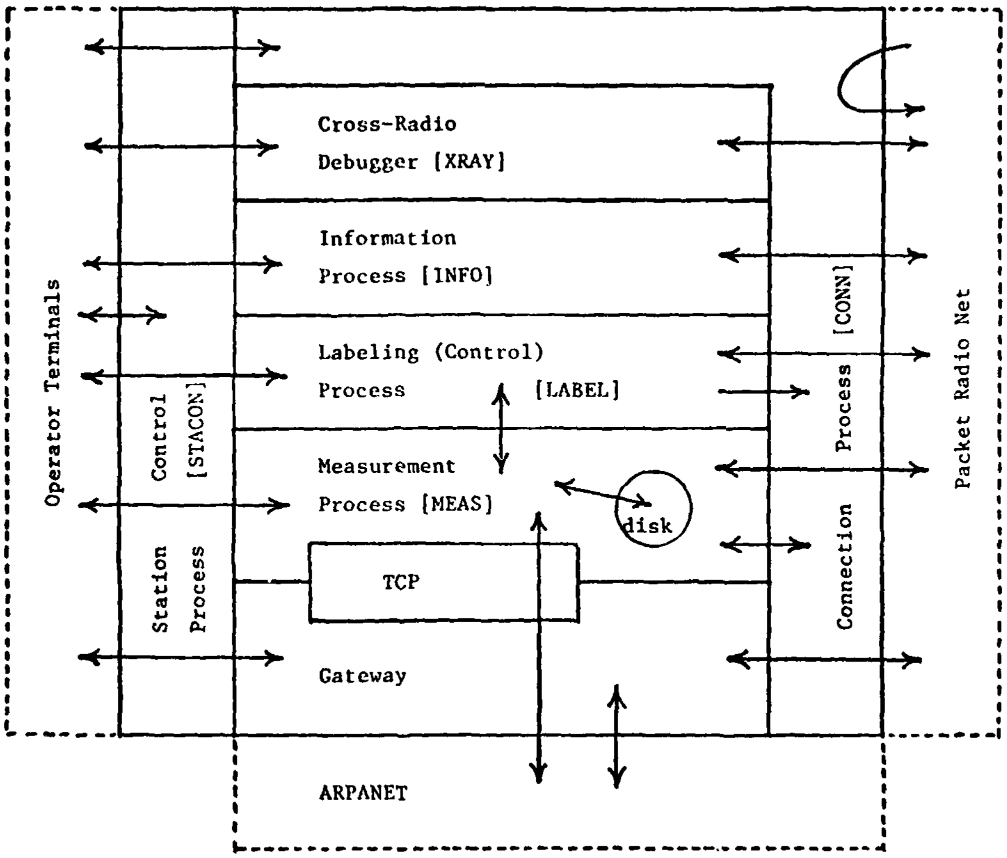
III. THE PRNET STATION

A major milestone this quarter was the delivery to SRI of the version 2 station software. Although software updates containing new features, bug fixes, and other improvements have been made numerous times since the first station release in July 1976, this release contains several new software modules in addition to changes in old ones. In this section we will describe software developments since last quarter's report, summarize the current station software, and discuss problems concerning station resources.

III. A. Station summary

The following diagram shows the station (solid lines) and those entities in the outside world with which it communicates (dotted lines). Each station application process is shown in a separate area; operating system software is not shown. Arrows indicate communication between their endpoints, with the communication following the path indicated by the arrow.

STACON interfaces operator terminals to those station processes which do terminal I/O. Output from CONN, XRAY, INFO, LABEL, MEAS, and the Gateway is directed by STACON to appropriate terminals. Additionally, XRAY, INFO, LABEL, and MEAS accept operator typein via STACON. CONN and the Gateway do not accept direct operator commands, but can be signaled by STACON to modify certain parameters as directed by the operator. STACON also



communicates with the operator on its own behalf, handling commands related to I/O control and other matters.

CONN interfaces XRAY, INFO, LABEL, MEAS, and the Gateway to the packet radio network by implementing SPP connections between station processes and other devices in the net. It also forwards packets from one PRNet device to another by readdressing packets received from the net and retransmitting them.

XRAY displays and alters memory in PRs and TIUs as commanded by the operator. INFO maintains a user-name/device-ID correspondence by processing commands from TIUs and the operator, responds to queries about this data from the same sources, and supports communication links between PRNet users and the station operator. However, the TIU code to communicate with INFO has not been implemented yet by SRI.

LABEL is responsible for assigning routes between the station and PRs and maintaining a terminal-ID/PR-ID correspondence. It tells this information to CONN, for its use in routing packets, and also displays information on network status in response to operator commands.

MEAS conducts measurement experiments as directed by the operator. This includes setting measurement parameters in CONN and LABEL, in PRs, and in TIUs, and collecting measurements from those sources. (TIU measurements will be implemented next quarter; all the rest is done.) The measurements may be sent with TCP to an ARPANET destination, or may be written on the station disk, then read back in and sent with TCP. The TCP communicates with the outside world via the Gateway. The Gateway accepts internet packets from the TCP, the PRNET, or the ARPANET and readdresses them to their destination, which may be on either net.

III. B. Software development

III. B. 1. Station processes

The major development this quarter was in the disk spooling and TCP delivery of measurement data. Last quarter we decribed the modifications made to the PDP-11 TCP supplied by Jim Mathis of SRI in order to run under ELF. At that time, the measurement process was capable of using this TCP to transfer measurement data, and this transfer was being tested by sending to a special test receiver in the station. We also described last quarter the redesign of the gateway that would provide for a good interface between TCP and the gateway. This quarter the TCP was integrated with the gateway, and a feature was added to it to simulate a TCP at the destination end so that it could be used to transmit to a raw packet sink, specifically UCLA's measurement data receiver. TCP transfer was tested both to UCLA (using the TCP simulator) and to the PRDATA program on a TENEX (see section 3 below), which receives packet radio measurement data via TCP and stores and prints it. The station TCP was debugged with Jim Mathis's help. A multi-step procedure was used to check that the data was properly received at UCLA. First of all, the measurement process was told to print all data it gathered and sent, so we had a typescript of the data. UCLA then sent us, in a computer message, a hexadecimal dump of the data they received. We filtered out duplicate packets from the listing, then converted it back from text into a binary file and used it as input to

PRDATA (see section 3 below), which printed it out as if it had been received and filed by PRDATA in the first place. Comparison of this printout with the original measurement process typescript verified correct transmission of the data.

Disk spooling of measurement data allows measurement experiments to be done at times when UCLA's receiver is not available and provides buffering to prevent backing up of data if the data is being transfered with TCP. When setting up a measurement run, the operator can specify a TCP destination (if any) and can enable or disable use of the station disk. The data is then handled as follows:

	TCP not in use	TCP in use
Disk disabled	flushed	sent directly with TCP
Disk enabled	stored on disk	stored on disk; all data on disk not previously TCPed is TCPed

Use of the disk provides some protection against interruption of data transfer due to failure of the destination. Data from separate runs is stored separately on the disk, and data from large runs is broken up into separately identifiable segments. Each run segment is transfered over a fresh TCP connection, and is not deleted (for overwriting with new data) until it has been completely transmitted. Thus if transmission of a segment is interrupted, the segment is preserved and retransmitted in its entirety next time. The measurement process provides several commands to allow the operator to deal with the data stored on disk. For example, the operator can find out what runs are stored there, can delete data that shouldn't be transmitted, can force data to be retransmitted, etc. All disk software testing was done using SRI's facilities via the ARPANET, as our disk was not yet available.

Additional software changes were as follows. Two features were added to XRAY, the cross-radio debugger, in response to requests from SRI. XRAY was changed to ignore any command line starting with ";" so that the operator can enter comments into the typescript. Commands were added to set the default input/output radix to octal, decimal, or hexadecimal. Formerly the default was always hexadecimal, so that all output was in hex and input was hex unless special characters were typed to indicate octal or decimal for an individual number. Although hexadecimal was the most useful radix for debugging PRs, octal is more desirable for debugging TIUs.

Although TIUs do not yet send commands to INFO, the information process, it was discovered that INFO was not quite compatible with the TIU SPP implementation. INFO expected the packet it received which opened a connection to contain a command. It would process the command in the first packet, send any necessary reply, and close the connection, thus ignoring any subsequent packets. This prevented a device from tying up access to the INFO service by holding its connection open. TIUs, however, send an empty packet to open a connection, then send their data in the next packet. INFO was thus changed to ignore the empty packet opening the connection and process the command in the next packet, answering it and closing the connection as before. A timeout had to be added to prevent the connection from getting stuck if the second packet never came.

The connection process and gateway do not implement operator commands. In order to affect their operation, the operator had to use the cross-net debugger to set parameters in the programs. The parameters available controlled output from the processes -- selecting and specifying the content of packet typescripts. These typescript parameters, and also some parameters defining numbers of buffers, can now be set with commands to STACON (the station control module). STACON signals the processes to change their parameters as specified by the operator. This makes it much easier to modify the parameters, since the modification can be done locally at the station, without debugging the station remotely.

The number of buffers available for several purposes in the connection process and gateway was reduced. The earlier approach to dealing with packet loss in forwarding and gateway traffic was to increase the buffering to a much larger amount than it had been. Resource constraints in the station have now forced a more careful approach to this problem. Numbers of buffers that had previously been made very large were reduced to an amount which still provided an acceptably low packet loss rate. This issue is discussed more fully in section C below.

In order to gain some free space in the station, the dialogue for specifying route formats was removed from the labeling process. In CAP3 PRNET routing, the format of a packet route is arbitrary; the bits indicating the route field for each level are not built in, but are defined by the station when it labels the net. During station initialization, the labeling process allows the operator to define the route format if the default format is not desired. This dialogue for setting up formats took a substantial amount of code. Since it was not being used (the default route format was always considered satisfactory) and will be obsolete when CAP4, with point-to-point routing, is released this fall, the dialogue was removed from the labeling process. The space gained by this deletion will be made available for implementation of TIU cumstats in the measurement process. Space constraints in the station are discussed more fully in section C below.

The control packet sent by the measurement process to initiate measurements in PRs was changed, as specified by Collins. The original control packet zeroed the PR cumstat buffer without checking that one existed, thus clobbering the PR's program if it did not have measurement software loaded. The control packet and PR software were modified to prevent this.

III. B. 2. Station operating system

Several minor changes and bug fixes were done in the station operating system. Several bugs were fixed in the ELF and new versions of the system were distributed. A change was made in the cross-net debugger's mapping of ARPANET addresses into Internet adrresses. Formerly, the low 8 bits of the 24-bit Internet address corresponded to the 8 bits of the ARPANET address. Now, the high 8 bits of the Internet address contain the 2 bit ARPANET host address right justified and the low 8 bits of the Internet address contain the 6 bit IMP number right justiifed.

Gateway software for BBN, UCL and NDRE was modified to use the new mapping between ARPANET and Internet addresses. In addition, patches installed in the gateways last quarter to send packets over the Satellite channel via SIMP fake hosts prior to installation of the BBN-Etam VDH line were removed. Installation of the VDH line between the gateway PDP-11 at BBN and Etam was finished this quarter. Use of the gateways and traffic generators at BBN and UCL to send traffic via the BBN-Etam line and the Satellite channel was demonstrated remotely by UCL to the participants at the Packet Satellite meeting at San Diego in August.

III. B. 3. Support programs

Two auxiliary programs have been written to aid in debugging and in performance evaluation. PRDATA is used to gather, store and print measurement data from a Packet Radio Network. SIQTST measures delays encountered by Internet packets as they travel through TOPS20 or TENEX, the ARPANET, and Gateways.

PRDATA may be run as a detached job on any TENEX which runs TCP. It will listen for a connection from the Measurement Process in the Packet Radio Station which uses TCP11 in the Station. Once the connection has been established, the raw, binary data is stored in a disk file on TENEX. At a later time this file may be sent to another host on the ARPANET using normal FTP, or it may be given to PRDATA as input as if it had been read from the network directly.

If PRDATA is run as a normal, attached job, it first prompts the operator for a source of data which may be either a TCP connection or a disk file. PRDATA may be instructed to interpret the data. Several levels of detail may be selected and the resulting report may be printed on the on-line terminal and/or stored in a (text format) disk file. This may be printed later or it may be transferred to another site for analysis.

SIQTST is a TENEX program which has two processes; one receives messages which have been sent by the other. The messages are normal ARPANET type 0 messages which contain as data

Internet measurement type packets. The data portion of the Internet packet in turn has space allocated for TENEX-style millisecond timestamps. The monitor on TENEX was modified to look for this particular style packet an add to it tmestamps at the following points: in the SNDIM JSYS as the packet arrives from the sending process and is placed on the IMP output queue, when the packet is started out to the IMP, when the packet has arrived back and is queued for the receiving process, and when the receiving process (in a RCVIM) dequeues the packet and returns to user mode. In addition the two user processes add their own timestamps, i.e., just before sending and just after receiving.

Experiments were run using SIQTST in five different configurations. First the packets were addressed locally to BBNA, where SIQTST was being run. It was found that all network delay could be accounted for by the time to cross the interface. Since BBNA was being run stand-alone, only the network delay was of interest.

The second configuration was using a loop-back plug on a different host port on the same IMP to reflect the messages back to BBNA. Again, interface crossing times accounted for all delay. The third experiment used the Internet gateway in the Pluribus IMP. The messages sent from BBNA were addressed locally to the Pluribus, but the Internet address specified BBNA. Thus the Pluribus would send the packet back to BBNA. The fourth configuration was using the PDP11 Gateway on a different port of the same IMP as BBNA is connected. In all of these arrangements, the network delays were accounted for by interface crossing times and an acceptable amount of delay in the program which was reflecting the packets.

The fifth experiment involved running the PDP11 Gateway on an IMP which was two network hops away from BBNA's IMP. The delay in this configuration left 487 milliseconds unaccounted for. This is currently attributed to contention for transaction blocks in the IMPs. In particular, the Gateway was running on IMP 40 which also supports the Network Control Center and is therefore quite heavly loade. Furter measurements must be made to completely analyze the source of this (unacceptable) delay.

III. C. Testing, tuning, and station resources

With the development of new station software, and the increasing demands placed on the station by the expansion of the network, allocation of resources in the station has become a significant problem. The ELF operating system code and associated buffers and tables currently fill the available space in the kernel address space (between the interrupt vectors and the bootstrap). Because of this constraint, the number of several types of resources provided by the ELF system cannot be increased. In particular, the station uses a large number of I/O request queue elements (IORQEs) and interprocess ports (IPPs) both for communications between station modules and between the station and the ARPANET and Packet Radio Net.

The number of IORQEs and IPPs cannot be increased as there is no additional table space available in the kernel address space to support these. In addition to the resource allocation problems in the ELF system, there are also resource allocation problems in station processes. The packet radio stations contain the complete 128K words which the PDP-11 is capable of addressing. All of the user address space is being used by station processes. We have made all reasonable efforts to compact user programs by running more than one station process in the same user address space. For example, the TCP and gateway reside in the same address space. The user address space constraint prevents the expansion of station processes in terms of either additional coding or more buffering or table space.

To provide the resources needed in the version two station, we reconsidered the buffering strategy in the connection process and gateway process. Formerly, whenever the station forwarder or gateway dropped packets, the number of buffers in use by these processes was increased by a large amount. The resulting numbers of buffers used by these processes was felt to be unnecessarily large and a needless drain on station resources. Accordingly, tests were performed to determine the number of buffers to use in these processes with an aim to striking a balance between resource needs throughout the station and resource needs within these processes. As a result of the tests new versions of the connection process and gateway process with new buffering parameters were delivered to SRI. In addition, buffering parameters in these processes were made settable be STACON.

A document explaining how to set these parameters for optimal station peformance was given to SRI. By changing buffering parameters, we were also able to reduce the number of resources required from the ELF and to produce a version of the station capable of supporting Packet Radio Net operations and measurements. We are continuing to investigate the resource allocation problem and hopefully will devise a more long term solution.

IV. INTERNETWORKING

IV. A. Transmission Control Protocol/program

TCP for the TOPS20 monitor was brought up for the first time in early July and worked well enough to be used to log into the TCP Telnet server running at SRI. Since that time the last bits of coding have been completed and extensive debugging has been done. The TCP Telnet server has been modified to run under TOPS20 TCP and has been operated successfully in limited testing sessions.

The TENEX TCP (and TCP20) were modified to use the new mapping of 24-bit Internet host addresses onto the 8-bit local addresses used by the ARPANET. Several bugs associated with timing out partially synchronized connections and handling of overly long messages were diagnosed and cured in the TENEX TCP.

IV. B. The host/SIMP protocol module

The host/SIMP protocol module is a module in the gateway PDP-11 that provides the network specific code to interact with the satellite IMP, and also serves as a multiplexor, allowing fake hosts and a gateway to all talk to the same SIMP.

This quarter we started designing this module. One of the specifications for the gateway machine was that it not drop packets. However, we identified several situations in which deadlocks could occur unless packets were dropped in the gateway machine or unless the protocol was modified to allow a host to refuse packets (as the SIMP does). We negotiated with the people designing the protocol and agreed that the gateway machine would have to drop packets. We will, however, "drop slowly," which means that instead of simply dropping a packet we will block input for a short time to give a traffic sink a chance to free up a buffer, and only drop if no buffer gets freed in that time.

Other issues in the design of the host/SIMP protocol module are lack of memory for buffers and the need for minimizing processing time per packet. Under the ELF operating system interprocess port reads are very slow. However, to insure that only a single read would be required for a packet, a full length (512 words) buffer would have to be allocated. We decided instead to read in a small amount of the packet, decide from the header how big a buffer would need to be allocated, copy the initial part read into a bigger buffer, and do a second read to read in the rest of the packet. We will choose the amount to be read in on the first read large enough so that most packets will not require a second read and copying. (If we choose the amount too small then we will usually have to do two reads. If we choose the amount too large we will waste memory allocating larger than necessary buffers to small packets, and waste time copying a large amount of data.)

Another arrangement we made for saving memory was to put the host/SIMP protocol module in the same address space as the gateway. This way the two modules could, to some extent, share buffers. Instead of copying packets from one module to the other, we will pass pointers.

V. HARDWARE

Several anticipated enhancements to the hardware status were brought to fruition this quarter. The most important is the upgrade of BBN Packet Radio Station PDP-11 number 2. Additional core memory, a new IMP11-A interface, a disk controller and drive, and a terminal line multiplexor are the major elements of the upgrade. The memory, allowing execution of the full complement of today's station software, and the disk, permitting fast reload of software and storage of measurement data, were especially helpful to receive. Our plans for configuration of this equipment were detailed in last quarter's report. We followed those plans, which expedited the hardware work to reconfigure the systems (number 1 and 2), and has achieved a highly practical research and development facility.

Also this quarter, the NDRE gateway PDP-11 was installed in Kjeller, Norway. Some initial problems with the IMP11-A interface on that system were resolved by DEC, with the help of NDRE personnel and with BBN taking a coordinating and advisory role in the maintenance effort.

This quarter we also reached agreement with SRI regarding choice of the Digital Pathways TCU-100 backplane/battery powered clock to provide a station time base with continuity across power down cycles. We ordered, received and installed one TCU-100 unit in each BBN station. We anticipate future station software will utilize these clocks to provide date and time information in various ways, such as timestamping measurement runs and providing time of day information to network users.

The final hardware item for this quarter is the negotiation of continued maintenance for the gateway PDP-11 systems at BBN and UCL. Since the llifetime of the project at UCL has been extended beyond initial expectations, this continued maintenance has become relevant. We have also agreed with division 6 of BBN that they will assume responsibility for maintenance arrangements for the BBN and UCL gateways, as appropriate, after the present maintenance agreement expires, as part of their Satellite net and internet work.

Bolt Beranek and Newman Inc.

Report No. 3645

Command and Control Related Computer Technology
Part II. Speech Compression and Evaluation

September 1977

Submitted to:
Defense Advanced Research Project Agency

TABLE OF CONTENTS

I. INTRODUCTION

During this past quarter, we developed a new model for generating the excitation signal for the synthesizer of the narrowband LPC vocoders, with the objective of enhancing the naturalness of the synthesized speech. Most present-day narrowband vocoders employ an idealized source (or excitation) model, which is either a sequence of quasi-periodic pulses for voiced sounds, or white noise for unvoiced sounds. This model seems to be largely responsible for the "buzziness" and lack of naturalness perceived in the resulting synthesized speech. Our new source model, called mixed-source model, combines both pulse and noise sources in a novel way. Based on the observation that spectra of voiced speech sounds (e.g., voiced fricatives and even certain vowels) exhibit devoiced or incoherent high frequency bands, the model divides the spectrum into a low frequency region and a high frequency region, with the pulse source exciting the low region and the noise source exciting the high region. The cut-off frequency that separates the two regions is adaptively varied in accordance with the changing speech signal. In Section II, we present the advantages of the proposed model over the pulse/noise model, and describe a method for implementing it. Synthesis experiments conducted using the above model with manually extracted cut-off frequency data indicate the power of the model in almost entirely eliminating the "buzzy" quality.

The new source model is general in that it can be employed in different types of vocoders such as LPC, homomorphic and channel vocoders, as well as in synthesis-by-rule applications.

A second topic that received considerable effort during the last quarter is objective speech quality evaluation. We developed several procedures for objective quality assessment of LPC vocoded speech. The results obtained from these objective procedures for five utterances processed by each of 22 LPC vocoders were correlated with corresponding subjective judgments previously collected in our subjective speech quality work. High correlation scores in the range 0.8 - 0.96 were obtained, which evidently demonstrates the validity and usefulness of our objective quality assessment procedures.

II. NEW ROBUST SOURCE MODEL

The commonly used source or excitation for the synthesizer of narrowband LPC vocoders is the result of an idealized model, which is either a sequence of pulses separated by the pitch period for voiced sounds, or white noise for fricated (or unvoiced) sounds. The major deficiencies of this model are two-fold: (1) Some speech sounds, e.g., voiced fricatives such as [z], are produced using both vocal cord vibrations and turbulent noise as excitation for the vocal tract; (2) Errors in the binary voiced/unvoiced (V/UV) decision are readily perceived by listeners as a severe degradation to the quality of the synthesized speech. To deal with the first deficiency, the source signal should be formed by combining voiced and fricated source signals in some manner. Previous attempts (for example, see [6]) on this "mixed" excitation have not been successful since the way in which the two excitation signals were combined resulted in a signal that was judged to be noisy. The second deficiency constitutes what we call a "hard-fail" effect on perception. It should be noted that the problem of making a reliable V/UV decision becomes increasingly difficult as the signal-to-noise ratio of the vocoder input speech is decreased. Thus, we define a robust source model as one which is adequate for <u>all</u> speech sounds and speakers, and which produces satisfactory results for a wide range of input speech conditions.

The idealized pulse/noise excitation model causes a machine-like speech quality, not characteristic of natural speech. In contrast, the residual-excited LPC vocoder is very robust and produces speech that sounds quite natural, but at a bit rate typically in the range of 10 to 16 kbps. We have started a task in the past quarter with the ultimate goal of developing a new robust source model that lies between the residual and pulse/noise excitation models; one that preserves as much speech naturalness and quality as possible but remains consistent with the narrowband requirement.

A. Mixed-Source Model

The new source model that we are currently investigating has both the voiced (pulse) source and noise source; it allows for selective excitation of different speech frequency bands by different sources. In particular, we are investigating a model where the spectrum is divided into a low frequency and a high frequency region, with the pulse source exciting the low region and the noise source exciting the high region. The cut-off frequency which separates the two regions is a parameter of the model, which is to be computed and transmitted to the receiver. The cut-off frequency is a continuous rather than a binary parameter, and we believe that it will have a "soft-fail" effect on perception in that perception will be fairly insensitive to small changes in its value. We expect that the data rate required for the transmission of the cut-off frequency would be small (about 100-200 bps), and would be more than compensated by the resulting improvement in speech naturalness and quality.

Below, we give two examples from real speech to demonstrate the simultaneous excitation of a low frequency region by a voiced source and a high frequency region by a fricated (or incoherent) source, even for some vowels. Figure 1 shows the spectra of the speech signal (over 5 kHz bandwidth) and of the LPC error signal for the voiced fricative [z] in "is" spoken by a male. A careful inspection of either spectrum reveals that the cut-off frequency or the dividing line between the coherent and incoherent regions is at about 650 Hz. A second example corresponding to the vowel [I] in the word "vicious" from a female speaker is illustrated in Fig. 2, which suggests a cut-off frequency of about 2 kHz.

B. Generation of the Excitation Signal

For fully voiced sounds (cut-off frequency CF=5 kHz), only a pulse sequence is used as in the pulse/noise model; similarly, for fully unvoiced sounds (CF=0), only a noise sequence is employed. For all other cases where CF lies between 0 and 5 kHz, both sequences are generated for a duration equal to the current pitch period, using a gain factor for each sequence that would be appropriate if it were applied alone as input to the synthesizer. (This means that both sequences will have the same energy, equal

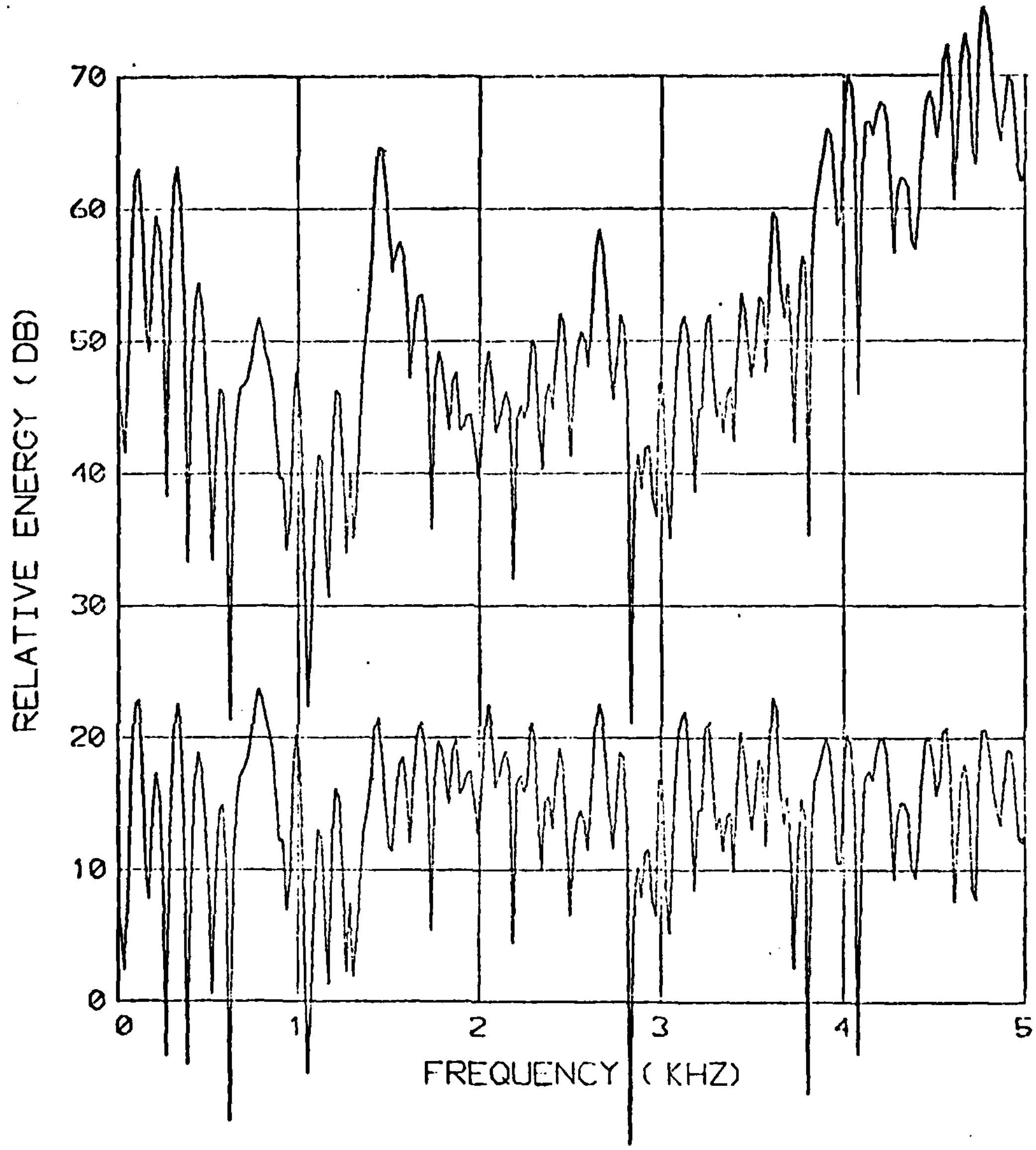


Fig. 1 Speech signal spectrum (top) and error signal spectrum (bottom) for the voiced fricative [z] in "is" spoken by a male.

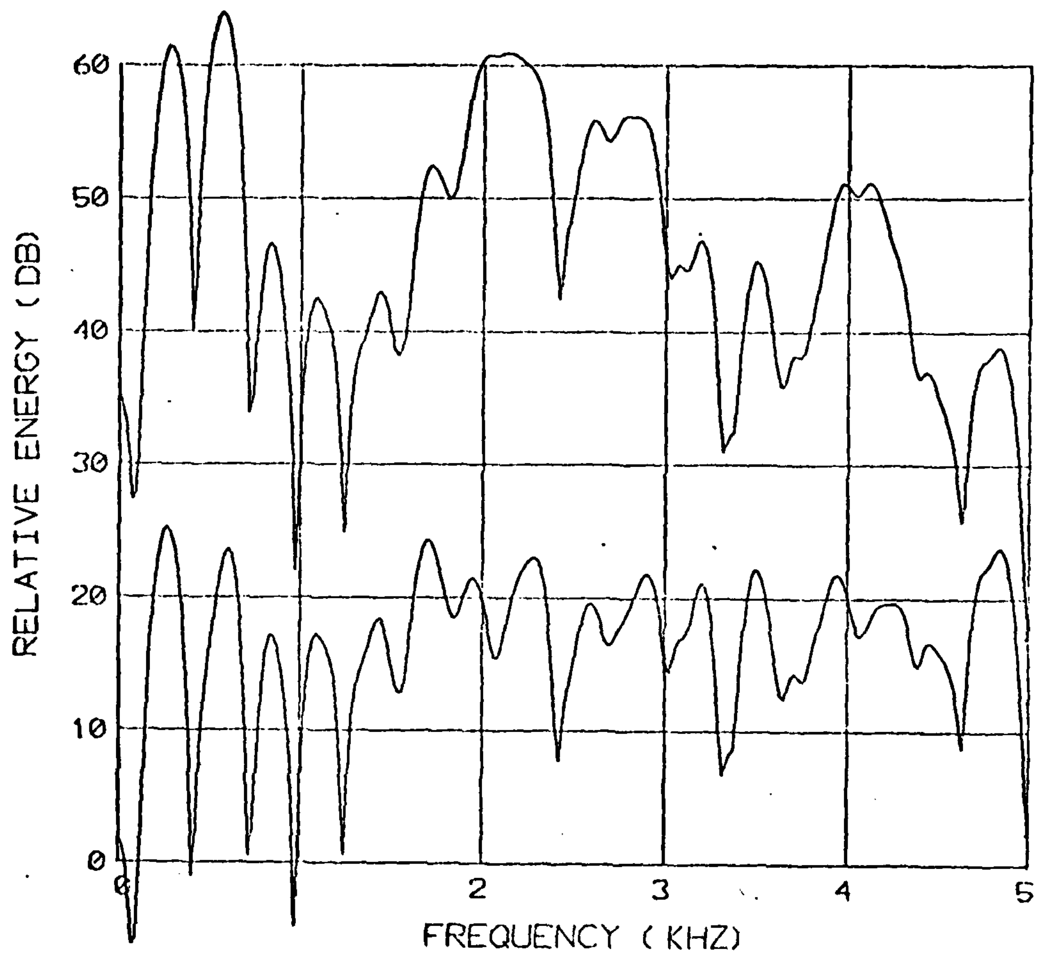


Fig. 2 Speech signal spectrum (top) and error signal spectrum (bottom) for the vowel sound [I] in "vicious" spoken by a female.

to the error signal energy.) The two sequences are then combined to form the excitation signal, as shown in Fig. 3 and explained below. The pulse sequence is passed through a low-pass filter with its cut-off frequency equal to CF, while the noise sequence is passed through a high-pass filter with its cut-off frequency also equal to CF. The two filtered sequences are simply added to give the synthesizer input. We use linear-phase low-pass and high-pass filters with a fairly gradual cut-off. Since the input sequences to the two filters have flat power spectra and the same energy, it is clear from Fig. 3 that the sum of the two filtered sequences should also have a flat spectrum and the same energy as either of the input sequences.

C. Manual Scheme and Experimental Results

To test the above mixed-source model, we conducted synthesis experiments using the model with manually extracted cut-off frequency data. For the purpose of this test, we chose the sentence RS3 ("His vicious father has seizures," spoken by a female) which has several voiced fricatives and which was found to have a noticeable "buzzy" quality when synthesized using the pulse/noise excitation.

For the manual extraction of the cut-off frequency, we developed an interactive display program on our IMLAC PDS-1 display facility. The program allows the user to observe, for a

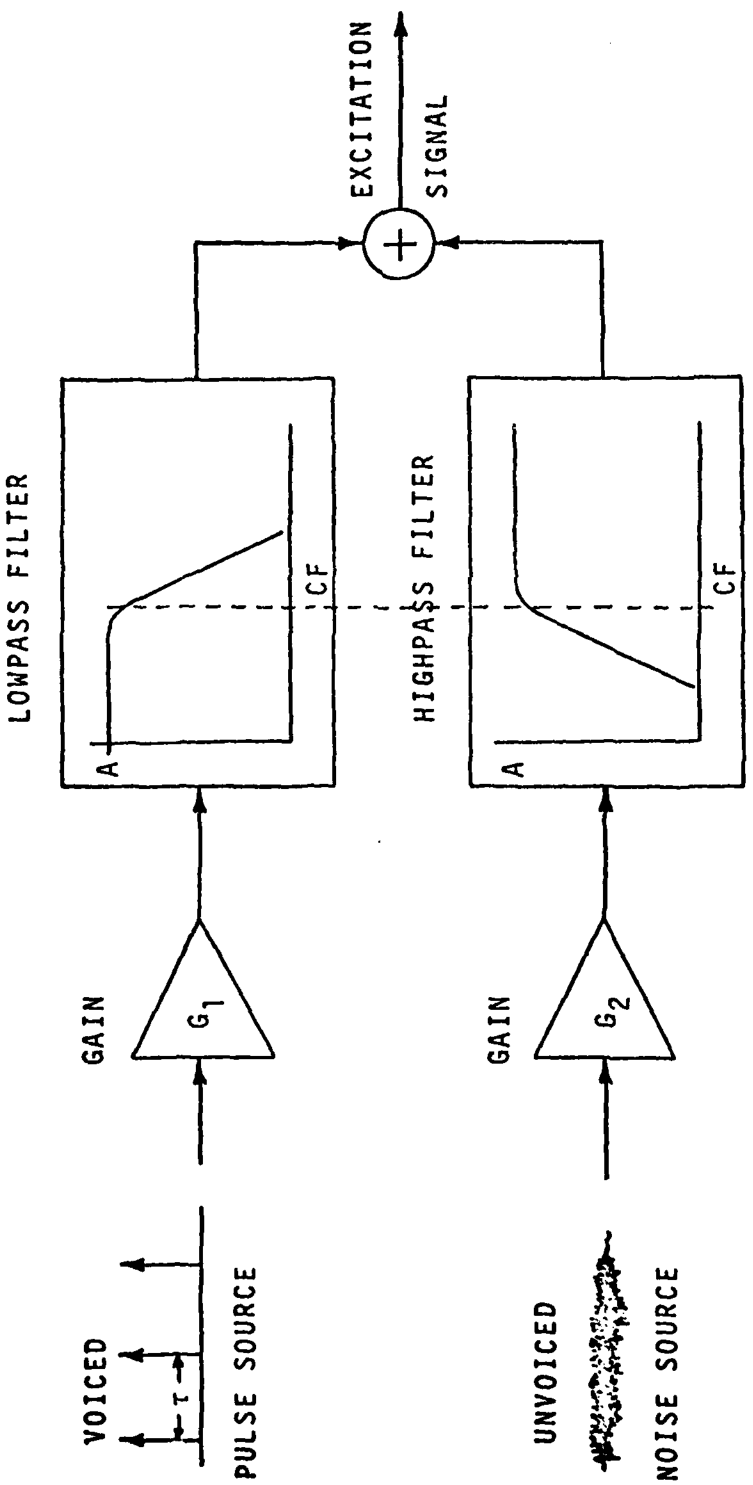


Fig. 3 New source model. (CF = cut-off frequency)

given speech frame, the waveform, the speech signal spectrum, and the error signal spectrum resulting from a 14-th order LPC analysis. The user may then enter the cut-off frequency value for that frame, and go on to the next frame. For voiced fricatives, the waveform is generally noisy and the spectra display the separation between the coherent and incoherent regions in a relatively clear fashion. However, for many other voiced sounds, it is generally difficult to decide where the spectral peaks cease to be periodic. To aid the user in this decision process, we incorporated a provision into the display program for displaying equally spaced vertical lines or marks along the frequency axis of the spectrum, with spacing equal to the pitch frequency estimate for that frame. The program read from a disc file the pitch estimates previously computed through our pitch extraction scheme based on the center-clipping method. The whole set of vertical lines could be shifted vertically or horizontally by turning appropriate knobs at the display terminal. The spacing between the vertical lines was changed either by typing a new pitch estimate, or by "dragging" one of the lines with a "light pen" so that it lined up with a chosen spectral peak. In the latter case, the program would compute the new pitch frequency estimate as the frequency of that spectral peak divided by the number of the vertical line that was dragged. With the display of the above lines, the problem of finding where the spectrum changes from a coherent state to an incoherent state is reduced to an easier task of observing when the peaks in the spectrum stop lining up with the vertical marks.

In the range 0-5 kHz, we considered only cut-off frequencies CF which are multiples of 500 Hz and the manually extracted cut-off frequencies were rounded to the nearest 500 Hz. The corresponding low-pass and high-pass filters were precomputed and stored away. We employed filters with linear phase and finite impulse response (FIR) properties. The filters were designed using Rabiner et al's computer program [7]. All the filters had the same order equal to 31, and had the same transition width equal to 600 Hz between the passband and stopband. The stopband attenuation for the different filters varied over a relatively small range from a low of 35 dB to a high of 38.5 dB.

For unvoiced sounds, we used a pure noise excitation (without any lowpass or highpass filtering). For voiced sounds, as noted in Section II-B above, we added the lowpass-filtered pulse sequence and the highpass-filtered noise sequence. A fully voiced sound corresponding to a cut-off frequency of 5 kHz would require the use of a pure pulse sequence. Since the FIR filters we used introduced a fixed delay at their output equal to 1.5 ms or 15 samples (half the length of their impulse response), a partially voiced frame followed by a fully voiced frame would present a pitch problem in view of the filter delay in the first frame and no delay in the second. (The pitch period in the first frame would come out shorter by 1.5 ms, which could be clearly perceived as a degradation especially for the utterance RS3 having a high fundamental.) To overcome this problem, we assigned a cut-off frequency equal to 4500 Hz for fully voiced frames. (An alternate solution is to "filter" the pulse sequence through a pure delay of 15 samples, for fully voiced frames.)

We resynthesized the sentence RS3 using the new mixed-source model with the manually extracted cut-off frequency data. The resulting synthesized speech had virtually no "buzziness" and sounded more natural than the same utterance synthesized using the pulse/noise excitation.

A careful listening test on the new synthesized utterance revealed the presence of occasional faint clicks. These clicks were also perceived in the synthesis from the pulse/noise model, although at a milder level. We believe that in this latter synthesis, the overwhelming "buzzy" quality has partially masked the perception of those clicks. We tried to investigate the source of the clicks by studying the synthesized waveforms and their spectrograms. We found that a technique where the usual pure voiced excitation of one pulse followed by zeros is substituted by a zero-mean sequence with a positive pulse followed by negative pulses all having the same amplitude (but with the same mean-square value as before), noticeably reduced the perceived level of the clicks.

Using the experience gained by the manual extraction of the cut-off frequency, we plan to implement an automatic algorithm that approximately mirrors the performance of the human experimenter.

III. OBJECTIVE SPEECH QUALITY EVALUATION

Before we report the work performed during the last quarter on objective speech quality evaluation, we first briefly review our previous work in this area so as to provide the necessary background for the problem.

A. Review of Our Past Work

We formulated a general framework for the objective evaluation of vocoder speech quality [1], based on the following reasonable assumptions:

(1) Speech synthesized from unquantized LPC parameters (14th order LPC filter, for a speech bandwidth of 5 kHz), extracted every 10 ms, is of very good quality, compared to the original speech.

(2) Except for pitch and gain, the fidelity of the short-time speech spectrum is the principal determiner of quality.

(3) The spectrum is uniquely defined by the linear prediction filter parameters.

The first assumption gives us an anchor point, defined in terms of the unquantized LPC parameters, against which to compare quantized realizations of the same utterance. The second and third assumptions relate the filter parameters to speech quality. In this framework, then, the problem of objective quality evaluation is reduced to the following two steps: 1) For each 10 ms frame, compute an objective error as the distance or deviation between the spectrum corresponding to the unquantized LPC parameters and the spectrum corresponding to the quantized and interpolated LPC parameters; and 2) Combine all the frame errors thus computed within a speech utterance into one number, which becomes the objective speech quality score. Notice that the described objective quality measurement procedure can be carried out when the LPC vocoder is in operation.

To perform the task of step (1) above, we developed several spectral distance measures which produced results consistent with published subjective perceptual results on formant frequency difference limens. A detailed description of these measures is given in [2]. Briefly, given two smooth spectra, the distance between them is computed in three steps:

(a) Normalize the two spectra by making them have either the same geometric mean (GM normalization) or the same value at zero frequency (DC normalization);

(b) Determine the error at each frequency as the magnitude of the difference in linear spectral amplitudes of the two spectra; and

(c) Compute the (weighted) norm of this error function after weighting the error with the perceived loudness function, originally developed by S.S. Stevens for a different purpose.

We chose to study in detail the use of two distance measures, denoted below as d(GM) and d(DC), which use, respectively, GM and DC normalization. In addition, we considered two other measures, d(RMS-LOG) and d(LAR), for comparative purposes; the first of these two measures computes the spectral distance as the rms value of the difference in the log spectral amplitudes of the two spectra, and the second measure is the Euclidean distance between the two p-vectors of LARs corresponding to the two spectra. Since LARs are readily available in the problem at hand, using the latter measure is computationally much less expensive than using any of the other three measures.

The task, in step (2) above, of combining the frame errors into one number involves first weighting the frame errors with a suitable time-weighting function to reflect the relative importance of the individual frames to perceived speech quality, and then averaging the weighted frame errors. Our work during the last quarter was directed towards developing specific methods for accomplishing this second task. Below, we describe these methods, and report on the correlation between the objective results and subjective judgments.

B. Time Weighting of Frame Spectral Errors

During the past quarter, we investigated the two time-weighting methods described below.

(i) Filter Gain Weighting: In this method, we make the reasonable assumption that frame errors in low energy regions of an utterance have a smaller influence on quality judgments than those in high energy regions. For example, even large changes in the spectrum may not be detected by the listener if the total energy in the spectrum is low. We considered the weighting as a function of the frame speech signal energy per sample expressed in decibels. A piecewise linear weighting function was found to produce good correlation between the resulting objective scores and the corresponding subjective test results.

(ii) Weighting Based on Our Perceptual Model: In the second type of (implicit) time weighting that we explored, we employed as anchor or reference our perceptual model of speech [3] instead of the 100 fps LPC analysis data. That is, we used the analysis data only for those frames for which our new perceptual-model-based automatic VFR scheme [4] decided to transmit; for all other frames, we obtained the LPC data via linear interpolation between the adjacent transmitted frames. In addition, we employed an explicit time-weighting in which frame errors for the transmitted frames are weighted with unity, while other frame errors are weighted with a fraction depending on the duration of the transmission interval to which they belong.

C. Time-Average of Weighted Frame Errors

There are a number of different ways of combining the weighted frame errors into one number. The simplest time-average is the arithmetic mean or straight average. We also considered a two-term composite average: the first term is simply the arithmetic mean over the whole utterance, and the second term is the arithmetic mean over the top 10% of the frame errors. A third measure we investigated is the above composite average but with the second term being a variable weight times the average over the top 10% of the frame errors; this variable weight was determined as an exponentially decreasing function of the "skewness" of the frame error distribution over the whole utterance. (The weighted second term in the composite average may be considered as the average over a variable percentage of large frame errors.) Note that (1) if the frame error distribution is skewed to the left, with relatively large numbers of small frame errors, then the skewness factor is generally positive, while (2) if the distribution is skewed to the right, with a relatively heavy concentration of large frame errors, then the skewness factor is generally negative. Therefore, the third time-averaging method described above weights the average over the top 10% frame errors with a larger factor for case (2) above than for case (1).

D. Correlation with Subjective Judgments

In our initial studies, we compared our objective speech quality scores against subjective test results obtained for the five utterances JB1, AR4, JB5, RS6, and DK6, and for 22 of the 48 vocoders included in our factorial subjective speech quality study [5]. We computed two types of correlation between the objective and subjective data: (1) regular, or Pearson's product-moment, correlation (we shall call this simply correlation); and (2) rank order, or Spearman's rank, correlation. For the second type, two sets of ranks are first assigned to vocoders under study using separately objective and subjective data, and then regular correlation is computed between the two sets of ranks. Correlation scores were used as a means of choosing the parameters of the time-weighting and time-averaging schemes discussed above.

Results obtained using the correlation study are briefly summarized below:

(i) Using the spectral distance measure d(DC) generally produced substantially lower correlations than using any of the other three measures investigated. Therefore, we eliminated the measure d(DC) in all our subsequent studies.

(ii) Correlation scores obtained for the utterances from male speakers were generally higher than those for the utterances from female speakers. Also, analysis of our subjective speech quality test results showed that subjective rating scores for the utterances from female speakers were relatively constant over the range of the number of poles (or LPC order) considered (9-14 poles); in contrast, the rating scores for male speakers exhibited a wide range of variation [4,5]. This suggested the variation of the LPC order for the anchor system as a function of the average fundamental (or pitch) of the speaker over the whole utterance. This technique was found to slightly enhance the correlation scores for the utterances AR4 and RS6.

ii) An important achievement of our objective speech quality evaluation work is that we obtained relatively high correlation scores. For the measure d(GM), correlation for individual utterances varied between 0.8 and 0.96; rank correlation had the range from 0.8 to 0.9. For the measure d(RMS-LOG), these ranges were found to be: 0.85 - 0.94 for correlation, and 0.83 - 0.88 for rank correlation. For the measure d(LAR), we obtained the ranges: 0.79 - 0.93 for correlation, and 0.78 - 0.83 for rank correlation. We plan to run the correlation tests over the full 48 vocoder systems employed in our subjective quality test.

IV. REFERENCES

1. J. Makhoul, R. Viswanathan and W. Russell, "A Framework for the Objective Evaluation of Vocoder Speech Quality," Proc. 1976 IEEE Int'l Conf. ASSP, pp. 103-106, April 1976. (Also, NSC Note 88, March 1976.)

2. R. Viswanathan, J. Makhoul and W. Russell, "Towards Perceptually Consistent Measures of Spectral Distance," Proc. 1976 IEEE Int'l Conf. ASSP, pp. 385-388, April 1976. (Also, NSC Note 88, March 1976.)

3. R. Viswanathan, J. Makhoul and R. Wicke, "The Application of a Functional Perceptual Model of Speech to Variable-rate LPC Systems," Proc. 1977 IEEE Int'l Conf. ASSP, pp. 219-222, May 1977. (Also, NSC Note 108, May 1977.)

4. BBN Quarterly Progress Report, Command and Control Related Computer Technology, BBN Report No. 3621, July 1977.

5. BBN Quarterly Progress Report, Command and Control Related Computer Technology, BBN Report No. 3520, March 1977.

6. F. Itakura et al., "An Audio Response Unit based on Partial Autocorrelation," IEEE Trans. Communications, Vol. COM-20, pp. 792-797, Aug. 1972.

7. L.R. Rabiner and B. Gold, Theory and Application of Digital Signal Processing, Prentice-Hall: Englewood Cliffs, N.J., 1975, pp. 187-204.