AN ON LINE DOCUMENT RETRIEVAL STRATEGY USING THE PLATO SYSTEM

ELISABETH R. LYMAN

Computer-based Education Research Laboratory

University of Illinois   Urbana Illinois

AN ON-LINE DOCUMENT RETRIEVAL STRATEGY USING THE PLATO SYSTEM*

Elisabeth R. Lyman

Computer-based Education Research Laboratory
and
Coordinated Science Laboratory

ABSTRACT

A preliminary experiment is reported using the computer-assisted instructional techniques of the PLATO teaching system for retrieval of bibliographic references linked to a data base model in the subject area of Coding Theory.

ACKNOWLEDGMENTS

AN ON-LINE DOCUMENT RETRIEVAL STRATEGY USING THE PLATO SYSTEM

Introduction

There are many computer-based information systems in existence or in the process of being developed. These systems offer considerable assistance to the researcher in need of information which has appeared in a myriad of professional journals and monographs. Existing retrieval systems, however, ordinarily require specialists who serve as an interface between the researcher and the computer because the operation of the systems require skill in computer retrieval techniques. The use of computer-assisted instructional techniques suggests itself as a means of facilitating information retrieval for the researcher or user unskilled in computer retrieval. This paper describes a preliminary application of CAI to the retrieval of bibliographic references.

The PLATO System

The PLATO computer-based teaching system provides an excellent tool for developing user-oriented strategies for document retrieval. The PLATO system[1] is a high-speed computer-based man-machine interaction system with a convenient visual display, keyset input device, and versatile programming capability including a simple programming language (TUTOR)[2], a time-shared on-line editing capability, and a logical program structure of interconnecting units particularly applicable to the "pathfinder" patterns helpful for retrieval. In addition, another software feature, a dialogue logic, developed for PLATO by Mr. Paul Tenczar, author of the TUTOR language, facilitates even further the design and writing of retrieval programs. Authoring in "dialogue" is very simple, computer-stored information is efficiently packed, and the dictionary of acceptable phrases and words, which can be expanded at any time, can contain conveniently designated synonyms and ignored words.

The Data Base

The subject area, Coding Theory, in which the example of retrieval was carried out, was chosen because a data base model for this area had been reported in a paper on a "Relational Structure for Document Retrieval in Coding Theory" by Nicholas Matthews Esser, Jr.[4] and an implementation of such a system had been described in a paper by Y. Chang.[5]

The model for the subject area of coding theory, as structured by Esser, was initially subdivided into seven steps which included all the coding theory topics from a decision to transmit information in coded form through some medium, to the construction, testing, and use of the actual system. Esser described the subject area in each step of the model by means of a tree hierarchy.

A real data base of document references retrievable by keyword or descriptor search was not used in the PLATO retrieval experiment, but rather, arbitrary index numbers were assigned to each point in the retrieval so that each reference request could be identified. For purposes of illustration, however, real references were obtained and appear at the appropriate places in the illustrative figures in this report as if they had been retrieved by an indexing or keyword search.

A Sample Retrieval

A PLATO retrieval program should allow a person searching for bibliographic references to make queries and get satisfactory responses from a PLATO terminal. Such a program has been written using the PLATO "dialogue" logic in which the document searcher can interact with the computer to determine paths leading to retrieval of relevant references. A description of a sample retrieval follows.

The user first specifies the general field of his search, in this case, Coding Theory, (Figures 1-3):
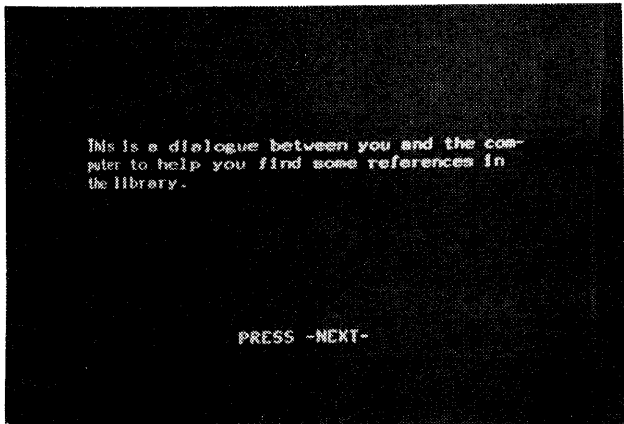
Figure 1
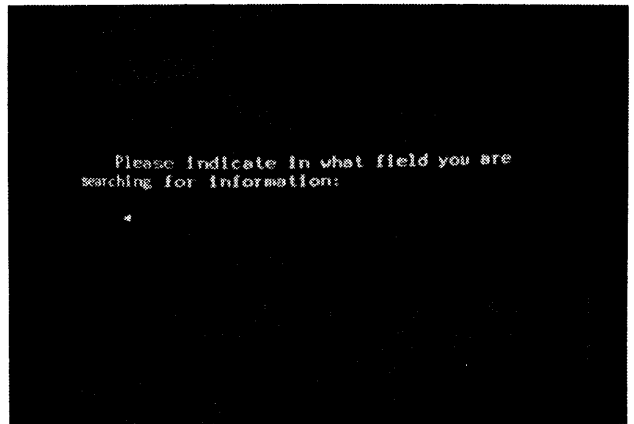
Figure 2
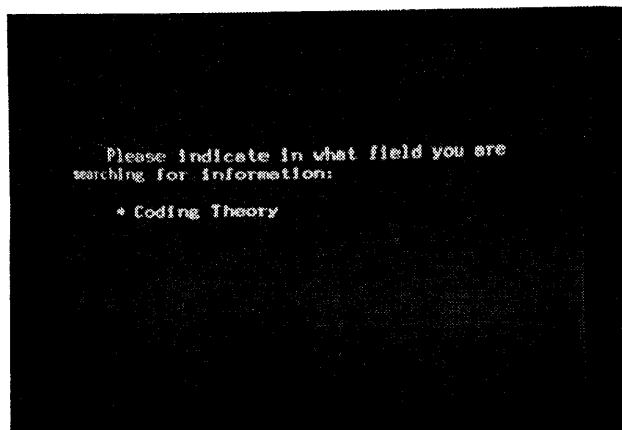
Figure 3

In order to pinpoint his request, he must first choose one of the seven

divisions of the topic (Figures 4,5):
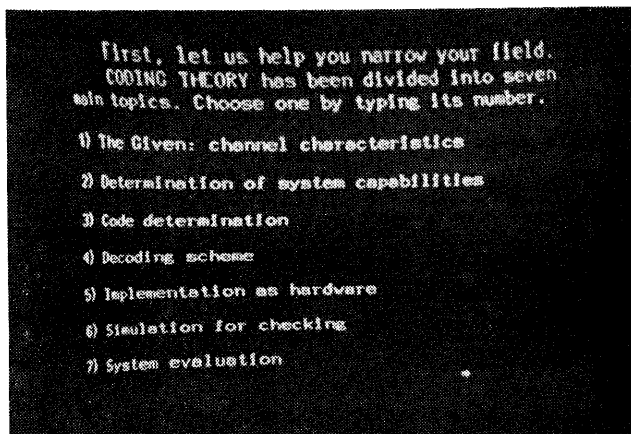
Figure 4
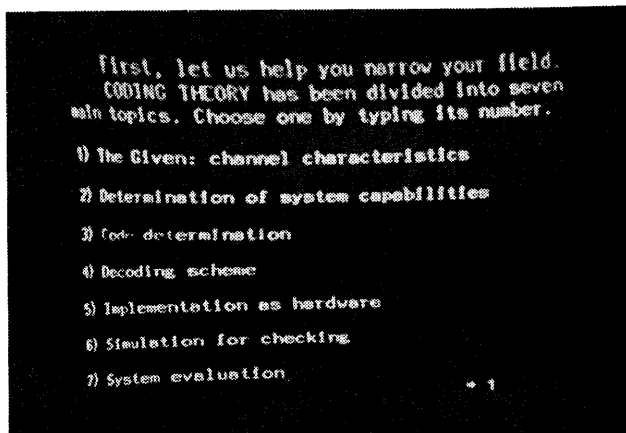
Figure 5

Having chosen topic 1 (channel characteristics) he next is instructed

to make a further selection of one of the subdivisions of the topic. He
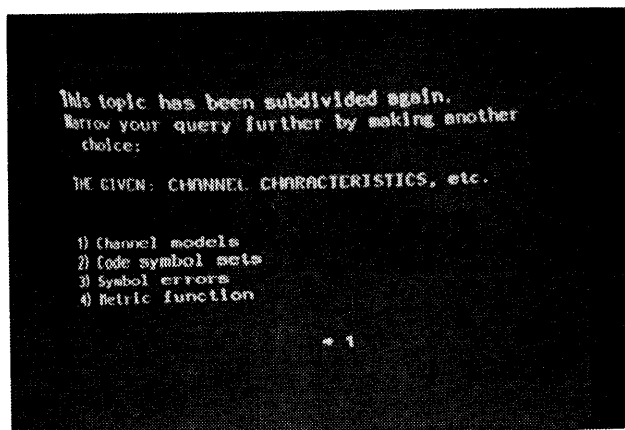
chooses "channel models" (Figures 6,7):
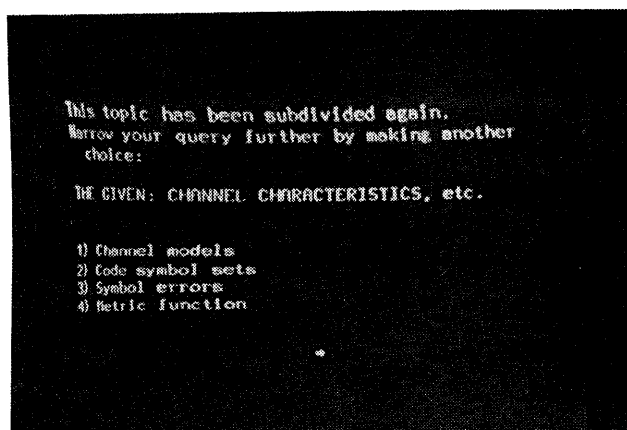
Figure 6

Figure 7

At this point he sees the path he is following appear in the upper lefthand

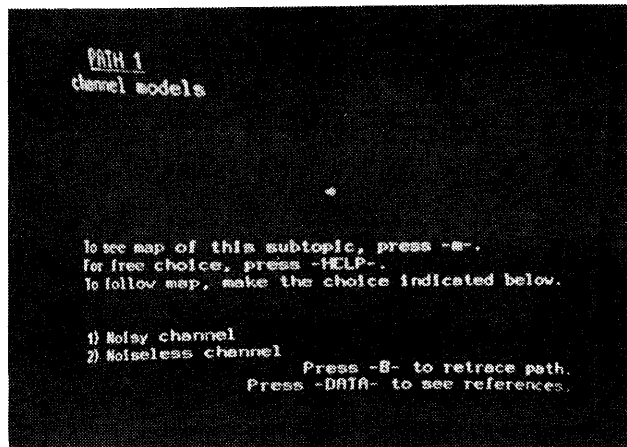corner of his screen, and he is given various alternatives from which to

choose (Figure 8):

Figure 8

Since he is unsure of his exact search topic, he decides to look at a map of the index hierarchy (Figures 9,10):
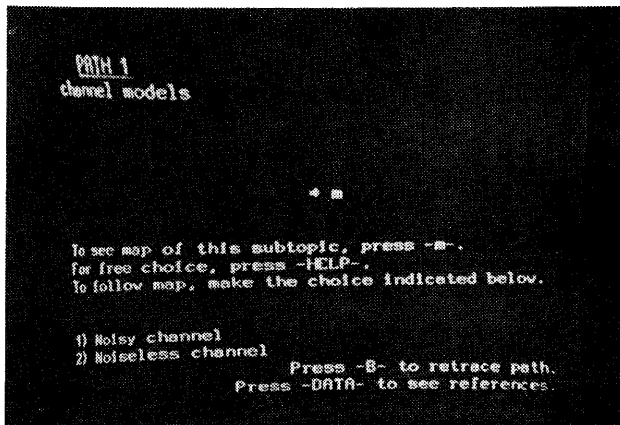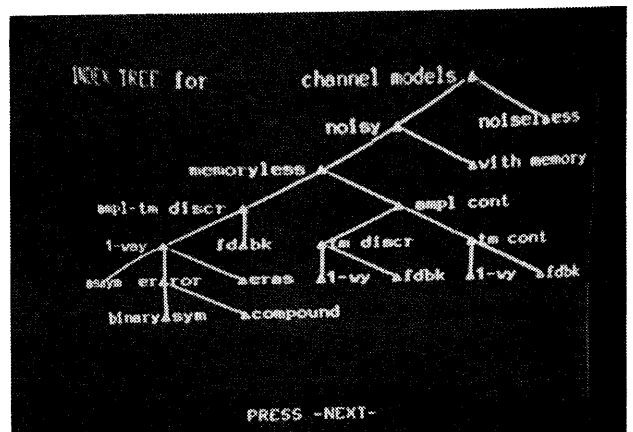

Figure 9
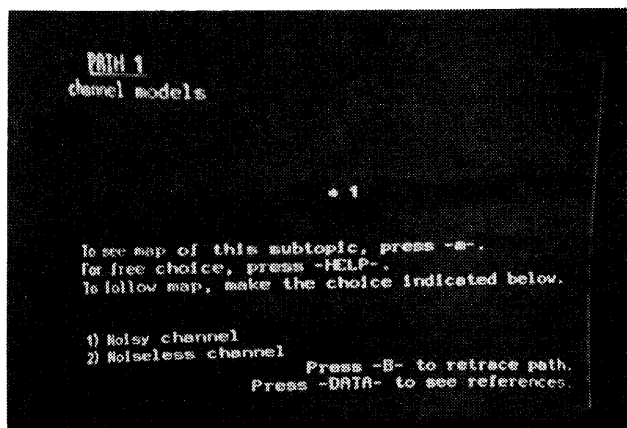

Figure 10

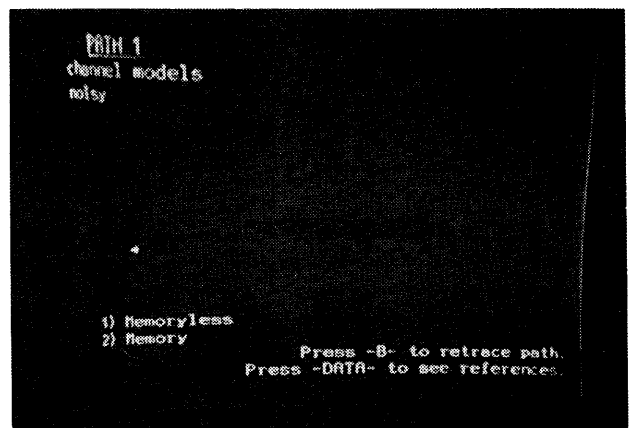He decides to follow the map (Figures 11-14):
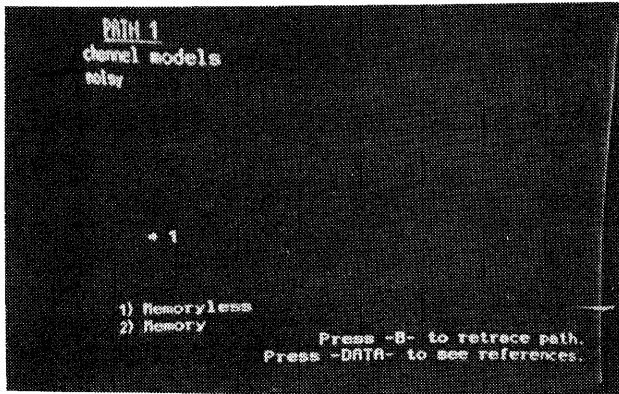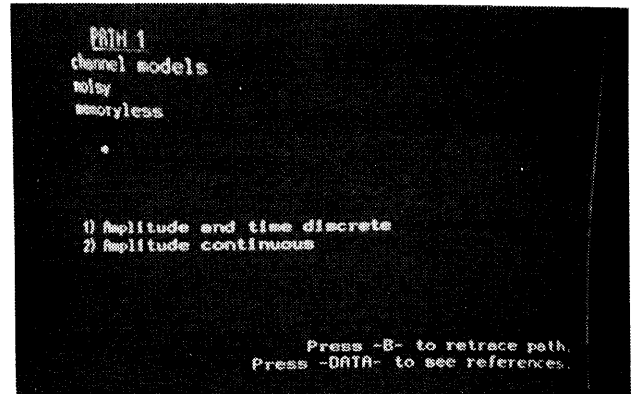

Figure 11

Figure 12

Figure 13

Figure 14

Having specified "memoryless noisy channels", he is curious about references available here, and so he presses DATA to see the list of references from which he can then take appropriate notes (Figure 15):
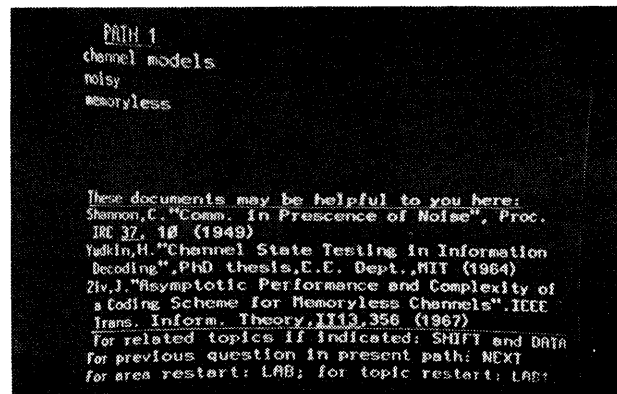

Figure 15

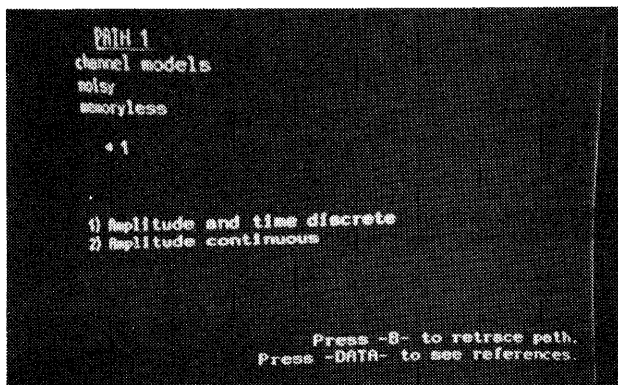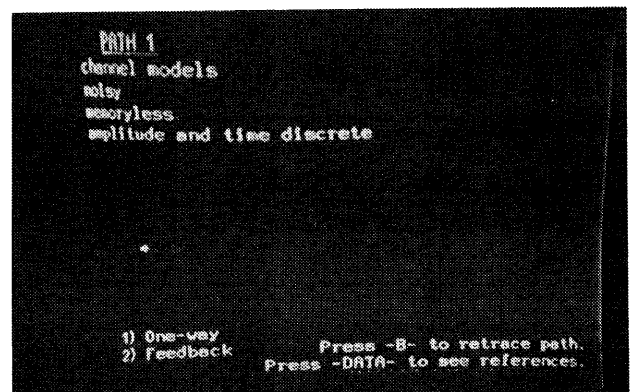The user decides to follow the map further (Figures 16-21). He presses NEXT, returns to Figure 14, types "1" and proceeds:
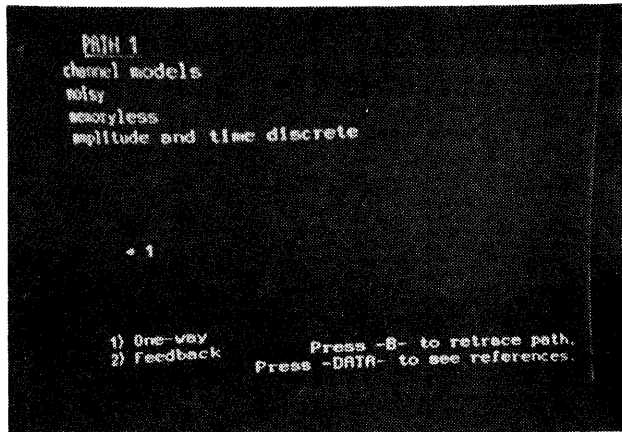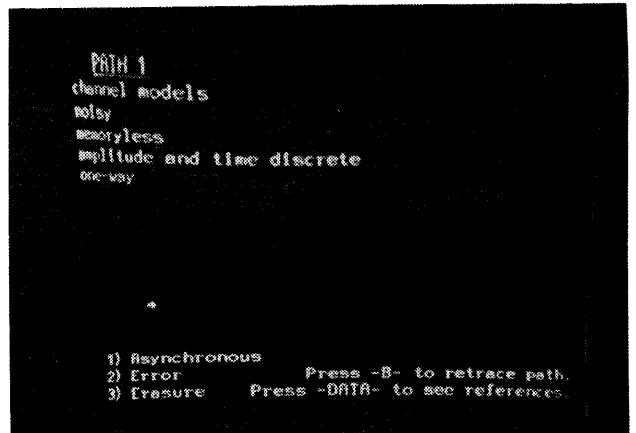

Figure 16
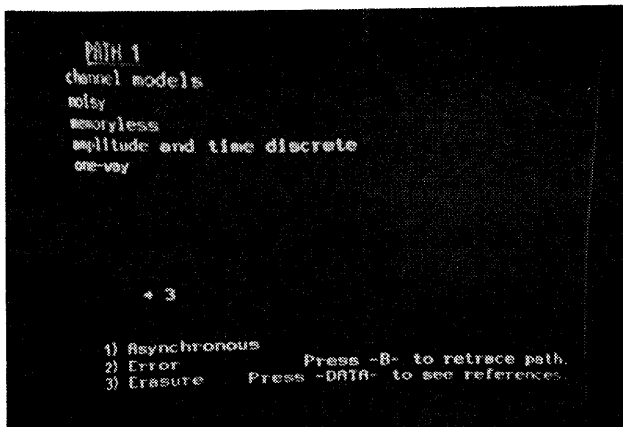

Figure 17

Figure 18

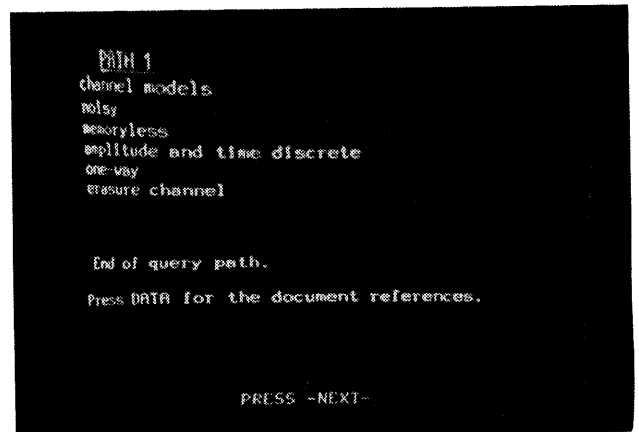

Figure 19

Figure 20

Figure 21

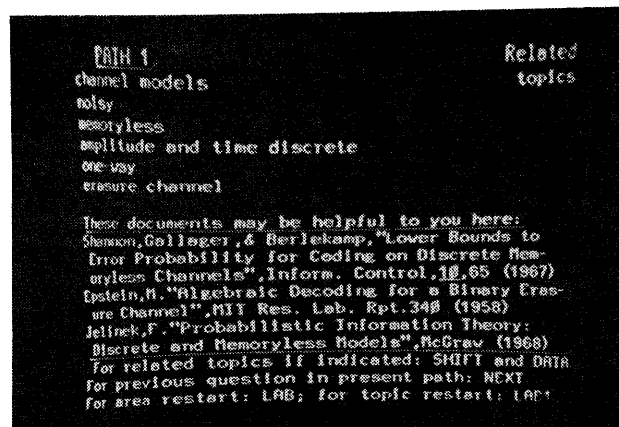Finding himself at the end of the query path, he presses DATA and again gets some references (Figure 22):

Figure 22

Here he notices a legend in the upper righthand corner of his screen:
RELATED TOPICS. The legend indicates he might find further references in
some other branch of the coding theory subject matter which also deal with
"erasure channels". He wishes to look at these references, so he presses
SHIFT and DATA and on his screen he sees the path he initially followed
(PATH 1) and the path he might have followed (PATH 2) to reach the references
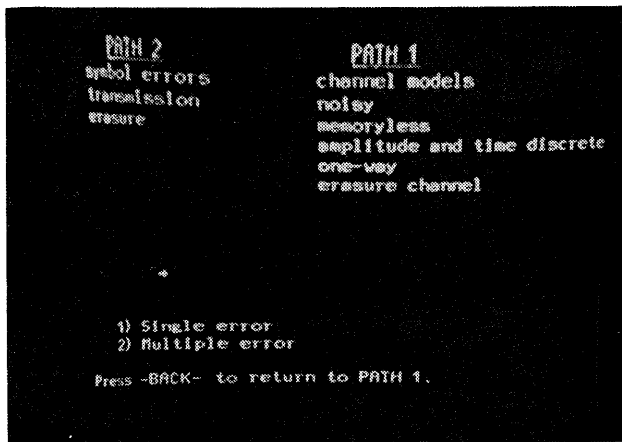to the related topics which are available here (Figures 23,24):

Figure 23
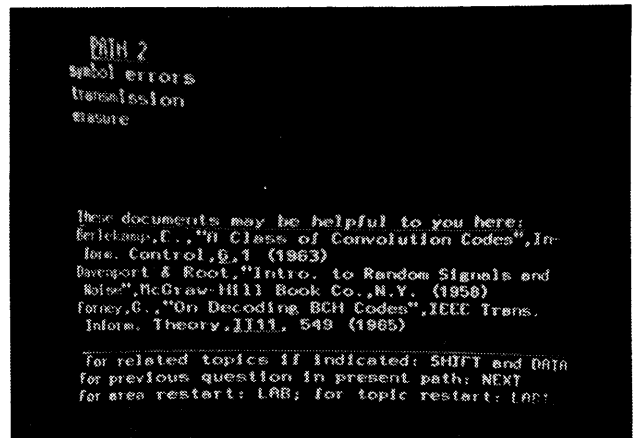

Figure 24

Since he recalls he was at the end of the query path in following the
"channel model" path (PATH 1, Figure 21),he decides to look at the subtopics
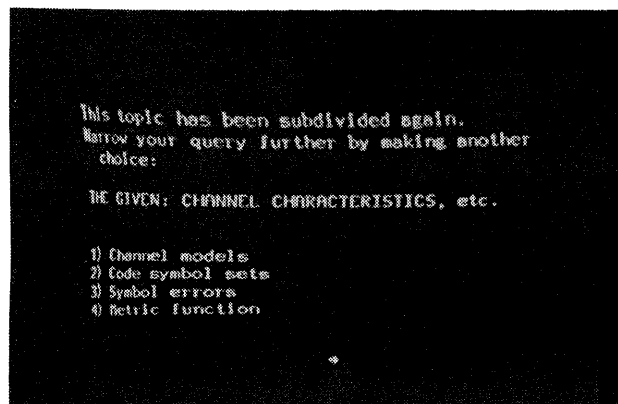in his subject area once more. He presses SHIFT and LAB (Figure 25):

Figure 25

Since the topics related to the user's last query could have been retrieved by following the "symbol errors" tree (PATH 2, Figure 23), the user out of curiosity looks at the map of subtopic 3 (Figures 26-29):
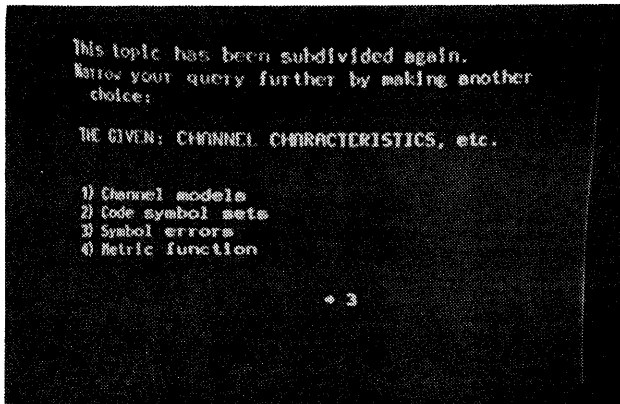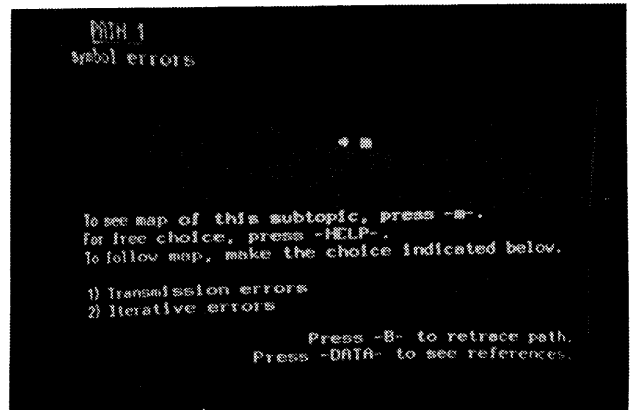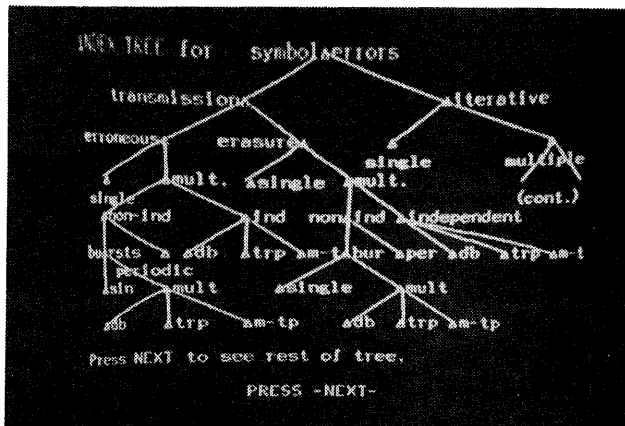

Figure 26

Figure 27

Figure 28
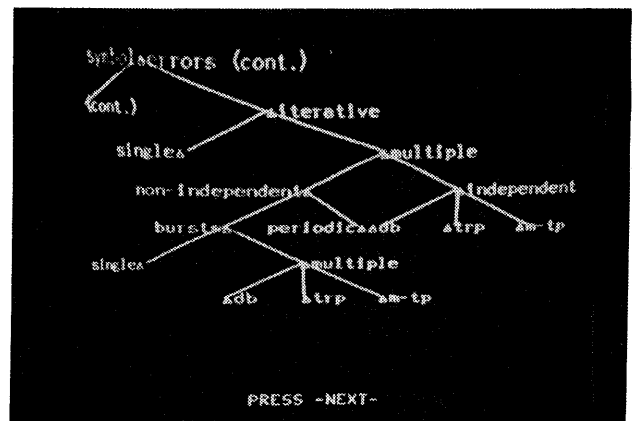
Figure 29

At this point he remembers that he had heard that J. Wolf had written articles on codes for errors which he would like to see, so he pushes the HELP button and types "Wolf references on codes for errors" (Figures 30,31):

Figure 30   Figure 31

The computer does not recognize his query as phrased, but he tries rephrasing

and this time can get references for this additional query (Figures 32-35):

Figure 32   Figure 33

Figure 34   Figure 35

Satisfied with the response to his "free choice" request, the searcher
tries two more such queries (Figures 36-38 and Figures 39-41):

Figure 36

Figure 37
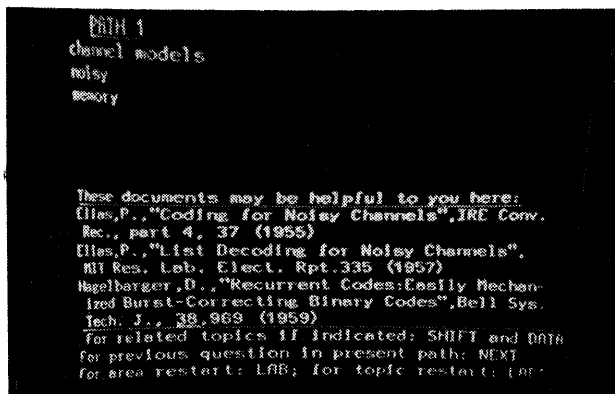
Figure 38

Figure 39

Figure 40

Figure 41

Discussion

The sample retrieval strategy illustrated in this report is one in which the query paths leading to a data base search are highly structured. The ultimate goal in retrieval as far as the user is concerned is the capability of response to natural language queries through some mechanism such as a computer system terminal. It seems logical, however, to work out some structured query techniques first. It should be noted again that no attempt has been made to determine the organization and search of a real data base. The work reported here is only concerned with an illustration of user query techniques.

The advantages of the PLATO system for user interrogation and retrieval are many. First the system is a rapid response system, and thus provides a time-saving facility for the user. Second, the PLATO "dialogue" is a convenient man-machine interaction program for user technique exploration which has even more simple authoring techniques than the usual PLATO TUTOR programming. Third, the dialogue program allows an easy expansion of a collection of users' questions to which the retrieval system should respond whenever users' recorded queries show phrases or words which were not originally included in the computer dictionary of terms and phrases. Fourth, in lieu of prestructured links of nodes in a tree structure, the PLATO programming could conveniently allow

appropriate algorithms, based on lateral and vertical relationships in the

tree hierarchy, to be substituted for precalculated linkages so that the

computer could generate query levels as well as links to related topics.

Fifth, the PLATO dialogue technique allows a history of events occurring

during the course of the interrogation to be stored in memory during query

which may then be used for subsequent modification of the retrieval strategy.

Finally, the advent of the new PLATO IV large scale system[3,6](1972-1975)

will offer inexpensive terminals available to many users and, although

PLATO IV will be principally dedicated to teaching, it could also provide a

convenient means for information retrieval in automated library systems.

Conclusion

It is hoped that the retrieval technique described in this report will

act as a catalyst for exploration of the applications of the PLATO system to

information retrieval.

BIBLIOGRAPHY

1. Alpert, D. and D. L. Bitzer, "Advances in Computer-based Education",
   Science, 167, 1582-90 (March 20, 1970).

2. Avner, A. and P. Tenczar, "The TUTOR Manual", CERL Report X-4
   (1969, revised 1971).

3. Bitzer, D. L. and D. Skaperdas, "The Design of an Economically Viable
   Large-Scaled Computer-based Education System", CERL Report X-5
   (February, 1969).

4. Chang, Y., "The Implementation of a Relational Document Retrieval
   System", CSL Report R-489 (September, 1970).

5. Esser, N. M. Jr., "A Relational Structure for Document Retrieval in
   Coding Theory", CSL Report R-469 (May, 1970).

6. Stifle, J., "The PLATO IV Architecture", CERL Report X-20 (April,
   1971).