1967

Grins, an on-line structure for the negotiation of inquiries

James Sproat Green V
Lehigh University

GRINS, AN ON-LINE STRUCTURE FOR THE NEGOTIATION OF INQUIRIES

by

James Sproat Green, V

A Thesis

Presented to the Graduate Faculty

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Information Science

Lehigh University

1967

This Thesis is accepted and approved in partial fulfillment of the requirements for the degree of Master of Science.

September 14, 1967
(date)

Donald J. Hillman
Professor in Charge

Donald J. Hillman
Head of the Department

Acknowledgements

The Author wishes to extend his thanks to Professor Robert S. Taylor both for his guidance and for the freedom which he permitted in pursuing this project.

Thanks are also due to Mr. A. M. Isaac whose ideas led to the search procedure incorporated in GRINS.

The work reported here was supported by grant from the Air Force Office of Aerospace Research, AF-AFOSR-724-66.

Table of Contents

List of Figures

Abstract

In general, problems are solvable along a continuum of abstraction. There is, at any given point in the development of the solution, a most efficient or optimum strategy.

In information retrieval systems the ultimate solution is obtained at a more specific rather than at a more abstract level. The question negotiation process is viewed as an efficient preliminary strategy which enables an information seeker to obtain his information goal with the least amount of overall effort.

In order for a problem solution procedure to remain efficient a means for predicting when to change strategies must be provided. In the particular example of question negotiation, this prediction is based on the rate at which the definition of the user's need develops.

An on-line computer program called GRINS is described which implements the information specialist's role in the negotiating of a user's need. This program communicates with the user in his natural conversational idiom. When the negotiation is judged by GRINS to be as well developed as it is likely to get, a search is made of the available documents. This search produces an ordered list of the sixty-three best documents which come close to the user's expressed need.

The structure of the program is modular so that improvements may be easily made. Some such improvements are suggested.

I. The Negotiation

When a researcher has an information need he has the problems of defining his need and of locating information sources which are able to satisfy the need. These problems require a strategy for their solution.

Problem solutions can, in general, be obtained at multiple levels along a continuum of abstraction. One may find solutions from the specific to the general.

Each such solution is obtained on the basis of a strategy which is characterized by the tools and techniques available at the particular level. The lower levels tend to have behavioristic or pragmatic techniques associated with them while the higher levels more often have generalizations which act to make transformations between various samples found at the lower levels.

The formation of a plan, followed by its execution, is an example of a multi-level approach. In effect there are two solutions. The plan formation offers a means of vicariously performing the execution of a task using symbolic techniques before the plan is actually employed. The work at the abstract level has as its chief advantage temporal and spatial mobility at low cost. The abstract solution is relatively unencumbered by physical restrictions of time and space. There is the constant risk at the abstract level of error. The symbols may prove to be inappropriate. The resultant solution of the abstract problem is therefore an estimate of solutions to be obtained at the lower levels.

For a given researcher with a given set of capabilities at his disposal there is an optimum strategy for each aspect of whatever problem lies before him. Optimum here means that adequate results are obtained for the least expense (time, money or intellectual). The once optimum strategy will reach a point where it cannot offer much more, where it becomes expensive, and where some other strategy rises to the position of being the optimum (if only by default).

How can an existing strategy be analyzed for its appropriateness? Every strategy has its rate of return on increment of investment. If the investment increment is constant for a given strategy, the best absolute measure of the quality of the strategy is its rate of return. When this rate of return drops to some minimum level we may define this point as an operational limit of the effectiveness of the strategy. Of course this does not tell us if any alternative strategies are as good or better than the existing strategy. A dilemma results because we will not really know which strategy is best until we try each such alternative strategy. The dilemma is resolved (or at least smoothed over) by retreating to a meta-strategy with the problem of strategy selection. The solution obtained from the meta-strategy is a prediction (presumably based on prior experience) of which new strategy to adopt.

Information retrieval processes in a library system may be viewed as a form of the problem solution process with multiple strategies existing along a continuum of specificity.

The retrieval of information requires the selection of search strategies which operate along this continuum. The actual material that contains the needed information will be specific, whereas the title, abstract, indices, and other referential means are more abstract. Beyond these are the even less specific reference services which rely on the highly abstract techniques of human discourse.

When a user approaches an information retrieval system he must travel from the more abstract levels to the specific levels if he is to utilize the system. The system includes the means of locating the most appropriate materials and if a user attempts to go directly to a specific material, he is attempting to bypass the system. As has been pointed out previously (1,2), a user will often approach the reference disk in a library with too specific a request for his information needs. This request must be negotiated to a higher level so that the reference librarian can present to the user the best compromise between what the user needs and what the library has to offer.

The user's information need, however ill-defined, first must be at least crudely defined within the user's space of ideas. This space is simply a convenient way of characterizing the user's private system of coherent (to himself) thought. In order to make a definition within this space it is necessary to get a "fix" within the space as a navigator does. While the navigator may use whatever stars and lighthouses that may happen to be available, he will prefer to use those that are orthogonal with respect to one another. Likewise, in order to define the location of an idea within a space of ideas it is desirable to utilize orthogonal coordinates for the purpose. The coordinates here may be thought of as the residuals of past experience which have been reduced to symbolic representations of those experiences. By means of this cross-referencing the ambiguity may be reduced.

The goal of a question negotiation process is to reach an adequate definition of the need, first with respect to the user's space of ideas and then with respect to the information retrieval system's "space of ideas". The retrieval system's "space of ideas" may be represented by the system's means of indexing.

It is desirable to make these two steps in as efficient a manner as possible in order that the reference service will not become too costly to the user.

The efficiency of the first step is largely a function of the skill of the reference librarian as a question negotiator. Not only must the user be set at ease, he must also be stimulated along those lines that will lead to a satisfactory definition of his need.

By a skillful selection of negotiation strategies at the conversational level, the reference librarian attempts to elicit from the user sufficient information to get a satisfactory picture of his need. In this role the librarian acts as the navigator attempting to establish orthogonal bearings which are transformable onto the librarian's chart (index system). The measures that the librarian obtains will not be reliable until they produce internally consistant results. The librarian will attempt to develop as many independent lines of approach as are necessary to obtain such consistency. The user may be asked what he needs the information for, how certain aspects of his question are related, and so forth.

The transformation of the defined need into the retrieval system's space is essentially a problem of interpretation. The user's expression of his need may not be adequately understood by the librarian (acting as a part of the information system). The communication channel may be faulty either due to noise or to a disparity of idioms between the user and the librarian. If the idioms have an insufficient common basis the conversation must be conducted at a more general inguistic level which is common to both user and librarian and/or the services of an intermediary translator must be used. In brief, if the user's professional jargon is beyond the understanding of the librarian, somebody will have to explain it to him.

Given that the librarian is able to understand the user's question, it is additionally necessary that the question be defined with respect to the information retrieval system.

The two definitions of the user's need (his own and the system's) do not have to be made independently. Assuming that the user approaches the retrieval system with an insufficiently defined need, both definitions may be derived by the same interactive process. In this way the system can help the user by concentrating on those coordinates which are represented in both spaces (or nearly so).

The iterative process of defining the user's need should be permissive. The librarian should not attempt to force the user to make or reject any definition. This holds even for definitions previously suggested by the user. Any such attempts may thwart the free flow of ideas coming from the user and hence restrict the development of the most satisfactory definition to the user. The user should be permitted to express a change of mind such that he can steer the librarian towards his need. This change may be a shift in subject area, in emphasis, or in scope. This is to say that the negotiation is not viewed here as a hierarchical procedure of simply narrowing down the space of definition. Rather, the process is seen as the identification and development of relevant dimensions of thought without regard to any hierarchical structure. Figure 1. illustrates, in a two dimensional space, the changing scope and locus of a user's hypothetical expression of his need as the negotiation develops.

Alternatively, the movement of the user's interest as mapped onto a hypothetical three term coordinate index space is illustrated in Figure 2. Notice that toward the end of the negotiation the user's expressed need tends to settle in one neighborhood and as a result, the distance that the user's expressed need shifts on each successive interaction grows shorter.

A negotiation is considered useful until it reaches a point of development where further effort becomes uneconomical. That is, the negotiation is continued until little more definition of, or change in the user's need, is anticipated.

The rate of change in the definition of the user's need may be used to determine when the conversational strategy should be replaced by the search strategy. In this manner we have a formal means of conducting a (near) optimum negotiation of the user's information need.

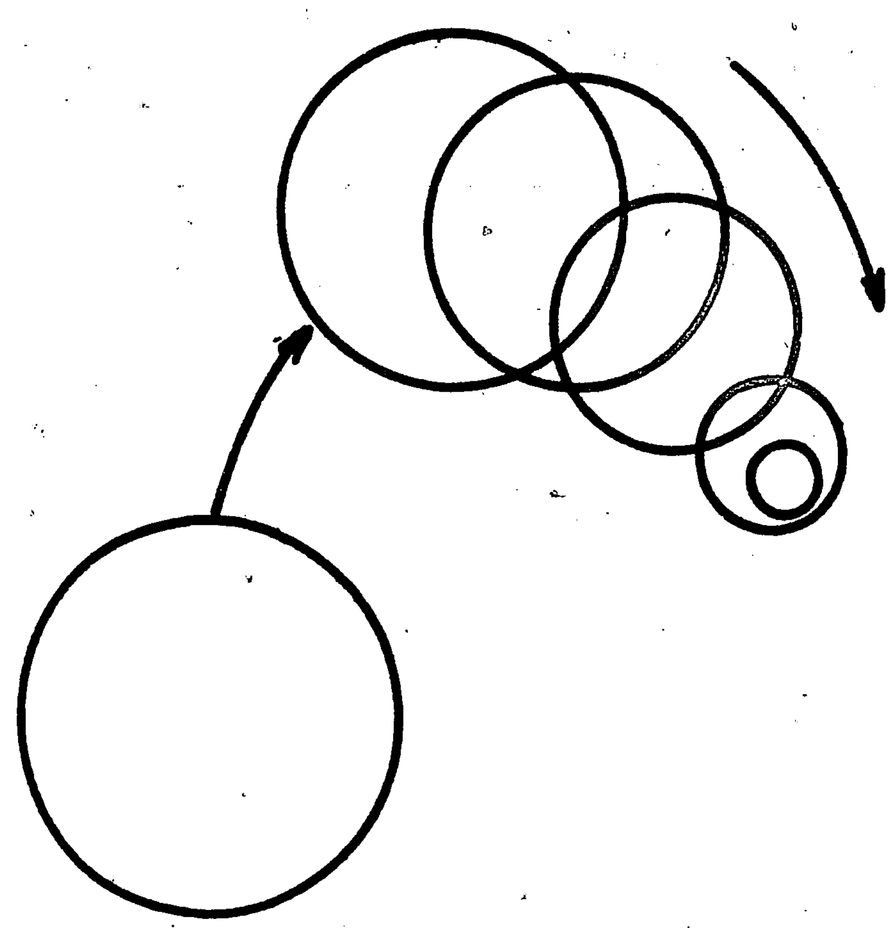

Figure 1. The Changing Scope and Locus of a Hypothetical Search Specification

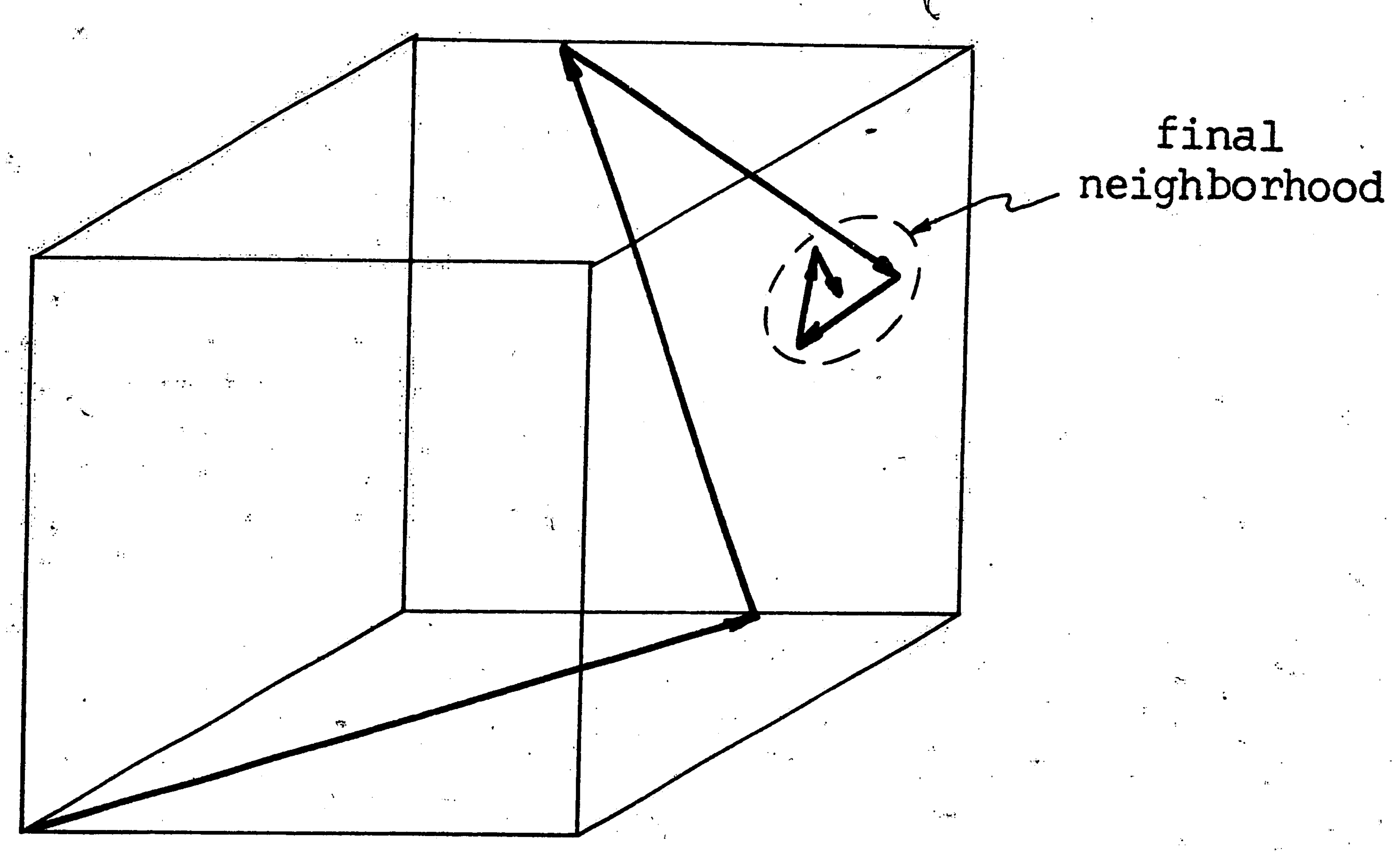

Figure 2. The Changing Locus of a Hypothetical Search Specification in a Three Term Coordinate Space

II. Implementation

In order to implement systematically the interpretation of the negotiation of a user's need as set forth in the first chapter of this thesis, several requirements should be considered.

Since the implementation is done by means of a mechanical device, certain restrictions are imposed on the dialogue that takes place between the user and the device. Communication will be restricted in both directions to forms that are machine generable and machine analyzable. This means that extra-linguistic forms of communication such as facial expression, tone of voice, and the like are not utilized. It would be convenient to imagine a user and a remotely situated reference librarian whose only means of communication are teletype machines. This restriction alone imposes a burden on the user, who must use a secondary communication modality in order to communicate. The user may be a poor speller or typist or both.

The pace of the dialogue must be necessarily a bit slower than a vocal dialogue because of the physical (not to mention the human) limitations on speed of character transmission. Furthermore, each party must wait his turn, as interruptions are not generally permitted.

The ultimate test of an information retrieval procedure is the extent to which it is used (after the novelty period wears off). This means that it must become more useful than competing, preexisting procedures. Its yield on user investment of time, effort, and money must be more favorable than other means.

In order to accomplish this the user must be met, as far as possible, on his own terms. If it were possible to "read" the user's mind by some advanced EEG or other technique, then this would be desirable since it would reduce the user's communication effort, however, for the present at least he is required to type out his responses.

It is necessary to place the user at ease in his communication with the reference librarian. The user is not to be distracted from his efforts in defining this information need. The confusion of a novice entering a system is especially dangerous here. A heavy usage of special formats and/or codes is to be avoided for this reason.

In brief, the user is to be permitted to use his normal conversational idiom. The requirements of the Queen's English (whatever they are) are considered to be too restrictive for these purposes.

A means for analyzing user responses must therefore be provided which will be quick, make few errors, and which has great tolerance of typographical and spelling errors and of grammatical variations.

The means by which the system provides feedback and stimulation to the user must be sufficiently flexible to allow for a wide variety of outputs. The general pattern of these outputs should at least appear to be relevant to the particular negotiation in progress.

Without the appearance of interest from the system, the user may quickly feel as though he is talking to himself alone (which is really what he is doing). Without the sense of an audience, the user may lose interest in his own talk and attempt to end the negotiation process prematurely.

In addition to the conversational capabilities a data base is necessary if a negotiation of the user's need is to be meaningful. This data base must be representable in an n-dimensional coordinate space so that the user's expressed need may be mapped into it.

A coordinate indexed collection of about twenty-six hundred documents is maintained at the Center for the Information Sciences (3). The documents are indexed with terms from a list of about 460 items.

Serial and inverted files are presently maintained for experimental and reference purposes. A computer search of these files is presently implemented by means of a formal search command which may include up to four numbers (representing index terms) which are combined with the logic operators "and", "not" and "or".

The formality of the present procedure is awkward for a potential user because: 1) he may not be able to frame adequately his need in the precise formality required by the system; 2) he may not be sufficiently familiar with the index terms used to index the collection; 3) his need may require more than four terms with differing degrees of emphasis on different terms; and 4) he may not have an a priori feel for the contents of the collection itself.

An interface is therefore desirable for this collection whereby a user can interact with the system on his own terms. Thus an implementation of a means to negotiate the user's need would serve the purposes of both the C I S collection and this thesis.

III. GRINS, A Working Model

A computer program has been developed* in order to provide a dynamic basis with which: 1) to simulate the role of the reference librarian in negotiating a user's information need; 2) for patterning more efficient negotiation strategies; and, 3) to act as a working interface to the experimental reference retrieval system of the Center for the Information Sciences at Lehigh University**.

This program, called GRINS 67 (for General Retrieval-Inquiry Negotiation Structure, 1967), is written mainly in FORTRAN II, with some details written in VFAP, (Valley Forge Assembly Program). GRINS was tailored specifically for on-line access using a teletype console and the computing facilities available at the Lehigh University Computing Center which include an 8K core memory GE-225 computer with four magnetic tape handlers, a disc storage unit and a Datanet-15 data transmission controller (for remote access to the computer), as well as other peripheral input-output equipment. Copies of the C I S system serial (document) and inverted (index term) files are maintained as a permanent store on disc. In addition a 'public' area is available for temporary storage on disc.

Because the core memory in this installation is severely limited, the program was written as a series of ten links, chained together on magnetic tape. The links, which are expandable to a total of 63, are

---

* Program listings are too lengthy for inclusion in this report. They will be available from the author's files for a period of five years.

** Referred to as the "C I S System".

modularized into functional units as shown in Figure 3. This modularization will permit later modifications to be made for the improvement of GRINS as working experience and experimentation dictate.

The chaining procedure is such that any link can be replaced in core by any other link. The common data area is left unaffected so that data stored by one link may be used by the following links. The following link is designated by the link currently in core memory. The ten links used in GRINS are briefly summarized in Figure 4.

These links may also be conveniently grouped as follows:

Conversation	Asker Listen Check
Connectivity	Asocat
Decision	Search Dump
Housekeeping	Init Found Termin Now

In addition to the permanent disc files mentioned above, several other files are used specifically by GRINS and are transferred from magnetic tape to the disc by Init at the start of the program.

These files include a list of referent terms, a list of basic responses that GRINS can make, a user information file and a control vector. Each of these files are explained in more detail below.

Several users may wish to use GRINS at the same time. While one user is negotiating his request another may be scanning some materials previously suggested to him by the system.

With GRINS the interaction process need not be restricted to a

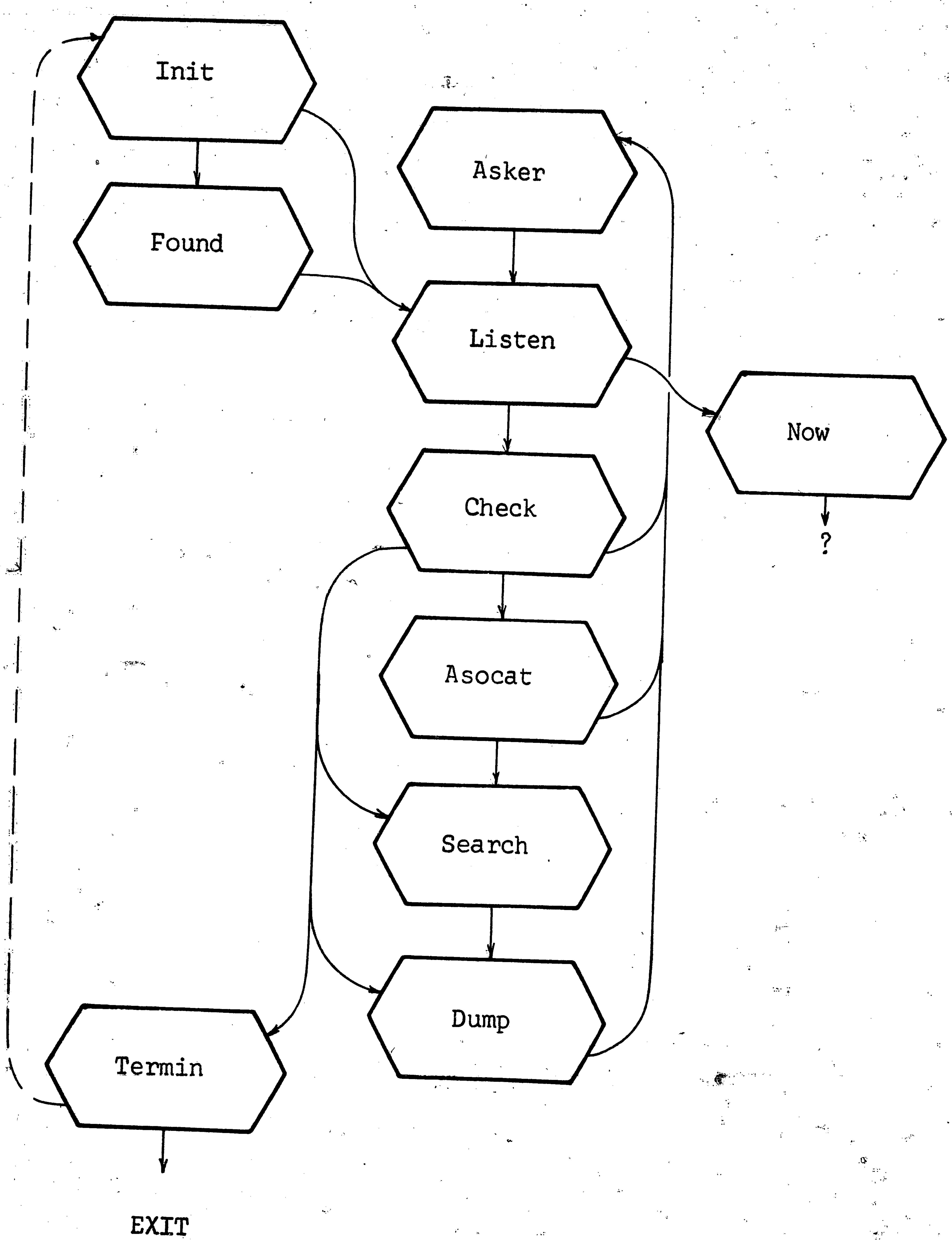

Figure 3. GRINS Links

Chain Link	Mmemonic	Function	Linked To
1	Init	Loads GRINS files from tape to disc Identifies user	3,10
2	Asker	Determine conversational mode Print GRINS' sentences	3
3	Listen	Analyze user's response	4,9
4	Check	Check analysis of response Control course of conversation	2,5,6,7,8
5	Asocat	Combine results of this and previous responses Test conversation for stationarity	2,6
6	Search	Form weighted-rank ordered list of documents	7
7	Dump	Present some document titles to user	2
8	Termin	Say goodby, prepare for next user or terminate program	1,EXIT
9	Now	Special link for modifying control vector	any link
10	Found	Extention of Init for previous users	3

Figure 4. Functions of GRINS links

single conversation. A user may return as many times as he wishes, to continue his searching. Features of his previous conversations can be saved for him in a user file set aside for the purpose.

A. Init Link

The Init link is always the first link to be called in GRINS. The current date is typed out on the teletype to document the conversation, followed by a greeting and a request for the user's name. If this is the start of the program, the files mentioned above will be transferred from magnetic tape to disc, and a notice that this is happening is presented to the user.

Regardless of whether this is the first or a later call to Init, the user's last name is checked against a user list of up to thirty-two names. If the name appears on this list the Found link is immediately called to replace Init.

If the name is not located, the user will be told that his name is not familiar. He is then asked if he has used the system before. If the user answers "yes", his name is rechecked. If the name is still unfamiliar to the system the user is asked:

WHAT DO YOU WANT, MR. _________?

The same title is used for all user's because of the lack of any simple method of inferring the user's sex, marital status or professional title.

The Listen link is then called so that GRINS can receive the user's first conversational response.

The identification procedure used here is embryonic and is currently justified only on the basis that the use of an individual's name tends to add a personal touch to his relationship with the system.

At some later date a brief set of operating instructions may be included here for those who may require it. In addition, the identification procedure can be elaborated upon if necessary for security purposes.

At the present time if two users have the same last name, one or both of them must adopt a pseudonym if they are to be distinguished. This is particularly pertinent if a user wishes to pick up his interaction where he left off at some previous time.

B. Found Link

The Found link is called only where a user has been recognized by the system. Currently, this link is largely a dummy. However, some of the terms that the user has made use of on previous occasions are loaded into a pushdown list for use by the Asker link. This link was formed to anticipate possible current awareness procedures which could be added at a later date.

In addition to including terms in the pushdown list, the Found link types the following message to the user:

ITS NICE TO HAVE YOU BACK AGAIN MR.________.
WHAT DO YOU WANT THIS TIME?

The Listin link is then called to replace this link.

C. Listen Link

In order to hold a dialogue with the user in his natural language, it is necessary to be able to understand the user's responses in some meaningful way. This requires that some analytic procedure be employed.

Several approaches are available for this purpose. Those considered for GRINS were syntactic analyzers, statistical techniques, and word look-up procedures.

The syntactic approach was rejected because of the complexity (if not the impossibility) of the task of designing an analyzer that is small enough to fit the available computer and at the same time rapid enough to analyze user responses within a few seconds. Furthermore, there was little guaranty that the user will in fact be formally grammatical in his conversational idiom. It was not considered desirable to impose such constraints on a user when we wish to make him at ease.

A simple word lookup procedure of matching term stems against words in the user response would be inadequate in the present application, because many terms used in the C I S index have common stems and/or are actually phrases, for example:

information
information seeking
information sciences
The Center for the Information Sciences

These are four separate index terms used in the C I S system. Also, there are many variations possible for some terms such as:

Association for Computing Machinery
Assn. for Comput. Mach.
A. C. M.
A C M
ACM

In addition some terms may appear with inverted orders:

information seeking
seeking for information

Any attempt to include so many variations in a list would be expensive in both file space and in time required to lookup each term. A lookup 'stop list' was excluded for many of these same reasons.

A statistical procedure was considered in which the relative frequency of alphabetic monograms appearing in index terms divided by the relative frequency of these same terms in the general English language was used as a basis for separating index terms from non-index terms. This procedure proved to be quite rapid and conservative of memory space and was able to achieve a 69% separation of the terms. This, however, was not operationally satisfactory (4). The use of digrams or trigrams would, perhaps, raise the degree of separation to a high enough level but many of the problems of the simple lookup procedure would still prevail.

The compromise solution which has been adopted is a modified lookup procedure which is capable of matching characters in specified combinations either within a single user response word or in a sequence of words.

The heart of this lookup procedure is a file of referent items (Figure 5) which correspond to index terms used in the C I S system. These referent items consist of key alphabetic characters of the index terms arranged in order, and, where the characters are not consecutive, an appropriate 'operator' is placed between them. For example, A+MATA# is the referent item for AutoMATA.

Any word that starts with the character "A" and which is later followed by the character sequence "MATA" before a space is reached will be interpreted as "AUTOMATA".

Likewise, LAT.CL#, which is the referent item for "LATent CLass analysis", will interpret any consecutive pair of user words that start with the characters "LAT" and "CL" respectively as "LATENT CLASS ANALYSIS".

Another operator occurs in the example, LIB*CONGR#, in which any sequence of words of which the first word starts with the characters

50100	AR+RIV	50200	ABS+ACTIN	50300	ABS+AC
50400	ACQ	50430	AC+NYM	50500	AD+APT
50600	AD+RES	50700	ADJECT	50800	ADVERB
50900	AER	-2	AFF	-1	AGREE
51100	ALGEBR	51200	ALGOL#	51300	ALGORI
51400	ALPH+B	-2	AL+RI+T##	51500	AMB
51600	A*I*C*E##	51690	ANALOG	51700	ANAL##
51800	ANS	51900	ANTHRO	51950	AN+NYM
51970	AP+LIC	-2	AP+ROV	52000	ARC+LO
52100	ARTICL	52300	A*C*M#	52400	AS+OC#
52600	ATOM.ENER	52620	AT+RIB	52710	AUTH.F
52700	AUTHOR	52800	A+MATA	52900	A+ATIC
53000	A+ATIO	53100	BASEBA	53150	BAT+PR
53200	BAYES#	53300	BEHAV#	53500	BIBLIO
53600	BIN	53600	BIT	53700	BIO
53800	BOOK##	53900	BO+LE#	54100	BROWS#
54300	BUSIN#	54400	CALC+L	54500	CAM+RA
54530	CANON#	54700	CARD##	54900	CATALO
55100	CATEG[SYN,STRU]	54900	CATEGO	55120	CRT
55120	C.R.T#	55250	C*I*S#	55200	CENT+R
-2	CERTAI	55300	CHAIN#	55400	CHARAC
55500	CHEM##	55550	CIR+TR	55600	CIT
55700	CLAS+F	55800	CLAUSE	55900	CLUMP#
56100	CORCL#	56200	COD	56230	COGNIT
56300	COMAC#	56400	COMIT#	56500	COM[CONT]
56600	COM+NICAT	56700	COMPAR	56800	COMPAT
56900	COMPIL	57000	COMPOU	57220	C.A.I#
57200	COMPUTER#	57100	COMPUT	57300	CONCEP
57400	CONCOR	57500	CONFER	57550	C+NECT
57600	CONSIS	57670	CONSTI	57620	CONSOL
57700	C+TENT	57805	CONTEX.FR	57800	CONTEX
57885	C+GEN.TAB	57900	C+TROL	58000	C+VERSION
58100	C+O+C+UR#	58200	C+OPER	58300	C+ORD#
58405	COPYRI	58400	COP	58500	COR+EL
-2	COR+CT	58600	COST##	58700	COUPL#
58800	CRANFI	58900	CREAT#	59000	CRITER
59100	CYB	59300	DATA.PROC	59200	DATA##
59400	D*D*C#	59450	DEACON	59480	DEBUG#
59500	DECIS#	59510	DECOMP	59600	DEFENS
59700	DEFIN#	59800	DEPEND	59900	DEPTH#
60000	D+CRIP	60100	DESIGN	60300	DIAGRA
57500	DIALO#	60400	DIAR##	60500	DICTIO
60600	D+RECT	60700	DISCO#	60800	DISPLA
60900	DI+SEM	61000	DI+SER	61200	DOC+AT
61100	DOC	-4	DOFSN#	61300	DUP
-2	DOUBTL	-4	DOUBT	61390	ECO
61400	EDG	61500	EDIT##	61600	ED[SUBJ,CUR,DIS,THEO
61700	EDUCAT	61800	EFF	61900	EIG
62000	EL+CAL	62100	EL+NIC	62200	ENG+R#
62300	ENVIRO	62400	EQ+P##	62500	ERR
62600	EURO##	62700	EVALUA	62800	EXPERI
63000	FACET#	63200	FACTO.ANA	63100	FACTOR
63300	F+DB+K	63500	FI+S+R	63400	FILE##
63700	FILM+X	63600	FILM##	-1	FINE##
63800	FLEXO#	63900	FLO	64000	FOR+GN
64200	FORMAT	64300	FORTRA	64380	FRENCH
64400	F+O+NC	64500	GAME##	64600	G*A*P#
64700	G*E*225##	64800	GEN+RAT+N	64805	GEN+RATIV
64900	G+ERIC	65000	GENETI	65200	GEOLOG
65300	GEOMET	65400	GERMAN	65500	GLOS+R
65600	GODEL#	-1	GOOD	65700	GOV+T
65800	GRAM##	65900	GRA[THEO]	66000	GRAPHICS#
65390	GRAPH#	66100	G+OP[DOC]	66200	HEAD##
66400	HEURIS	66500	H+ARCH	66600	HISTOR
73400	HAND##	66700	HOMOGR	67000	HUM+IT
66900	HUM	66800	HOSPIT	68500	IM.CON
68700	IND+XI	68600	INDEX#	68800	INDUCT
68900	INDUST	69000	INFER#	69300	INF[SEEK]
69200	INFO[SCI]	69100	INFO##	69400	INFORMAL#
69450	IN+OVA	69500	INORG#	69600	INPUT#
69700	INSTRUCT#	69900	INTERFACE	69950	INTERNAT#
70000	INT+OG	70100	INT+RV+EW	52200	INTEL[ART,SYN]#
69850	INTEL+GEN	70400	IPL	70420	IT+RAT
70450	JOV	-3	KNOW[DON,NOT]##	70500	KER
70600	KEYPUN	70700	K*W*I*C##	70800	L*REPR
70900	LAB	71000	LANG[ART,SYN,FORM]	71100	LANG+G
71200	LAT.CL	71300	LAT+CE	71400	LAW
71400	LEGAL#	71500	LEARN#	71520	LECOM#
71530	LEX	71700	LIB*CONGR	71600	LIBRAR
71800	LIM	71600	LINE*FORM	72000	LINEAR
72100	LING+S	72200	LINK##	72300	LISP##

Figure 5. Referent Items and Term Numbers

72500	LIS[PROC]
72800	LOGIC#
72900	MACHIN
73300	MANPOW
73100	MAN
73650	MATCH#
73900	MATRI#
74200	MEDLAR
74400	MEM
74600	METHOD
74900	M+CARD
75600	MORPH#
75800	NET
76010	NEUTRO
-5	NO
76300	NOMENC
-4	NOT ##
-2	OF.COU
76650	ON.LIN
76725	OPT+ZA
77000	OUT
77090	PARAPH
77300	PAT+ER
-2	PERHAP
77800	PETRO#
78100	PHONET
78500	PHOTON
78800	PHYSIC
79000	PLAST#
79340	PREFIX
79600	PROBAB
79800	PROCES
80500	PR+SIT
80800	PUBLICAT#
81100	PUS+WN
81400	Q+ES+N
81500	RANDOM
81900	READ##
62100	RECOGN
82400	REF
82600	RELEVA
82800	REQ
83000	RET
83300	ROLES#
83600	RUS
84300	SCI+ST
84500	SEARCH
84800	SEMA[COD]
85100	SERV##
80300	SHAR[P+J]
85400	SLIP##
85820	SOL.ST
85800	SOCIOL
86000	SPECIF
86300	STAT##
86600	STORE#
86900	STRUCT
87200	SUF+IX
-2	SWEL##
87600	SYMBOL
87900	SYNTAG
88200	SYSTEM
88500	TAP.TY
88700	TELEGR
88800	TTY
89200	TEXTIL
89400	THERMO
-4	T+NK(DON,NO)
89700	THRE+D[L]
90000	TIT
90250	TRAC##
90500	TRA[COMP]
90600	TRE
90900	U+D+C#
91200	URB
91500	VAL
91600	VIS
91900	WATER#
92200	WORD##
-1	YES ##
0	

72400	LIST##
72850	LOOKUP
73000	MAGNAC
73400	MANUAL+Y#
73500	MAP
73700	MAT+ER
74000	MEASUR
80200	MAC[PROJ]
74500	METAL#
74700	MICRO#
75200	M*I*R*F##
75700	NEGAT#
75900	NEURON
76100	NEWS##
76200	NOD
76433	NORM+Z
76600	NOUN##
-2	OK
76700	OP+RAT
76800	ORG+IC
77050	OV+L+D
77100	PARS##
77400	P*A*BO
77600	P+D+CA
77900	PHARMA
78200	P+COMP
78600	P+REPR
78900	PHYSIO
79200	P+DICA
79410	PREPOS
79700	P+R[SOLV]
80100	PROGRA
80600	PSY+LI
80900	P+CHED
81200	Q+ALIT
81430	Q+KTRA
81700	RANK##
81930	REAL.TIM#
82200	RECURS
82450	REGRES
82640	REM.AC
82905	R*D
83100	REV
83200	ROL
83800	SCAN[COL]
84100	SCIEN#
84550	SEGMEN
84700	SEMANT
85200	SET
85290	SIMPL#
85500	SMALL[GR]
80010	SOC[PROF]
85900	SORT##
86100	SPE+CH
86400	STER+D
86700	STRAT#
87000	STY
87300	SU+MPO
87400	S+ITCH
87700	SYNON#
88000	SYNTA#
88300	TAB
88400	TAP
88800	TELETY
89000	TERM+X
89100	TEXT##
89500	THES+R
89600	THINK#
89800	TIM.SH
90100	TOPOLO
90300	TR.FOR
90400	TRANSL
90700	TURING
91000	UNITER
91400	USER##
91540	VEC
91700	VOCAB#
92000	WEST.RES#
92300	WRI
92400	ZAT
0	

72700	LIT
73105	MAN*MACH#
73200	MANAGE
73401	MANUAL
73600	MA+KOV
73800	MATH##
74100	MEDIC#
74300	MEET##
74510	METEOR
74800	MIL
75400	MODEL#
57500	NEGO##
76000	NEUROP
-1	NO.DOU
76250	NOISE#
76500	NOTAT#
76620	OBSOLE
-2	O.K ##
76720	OPTIC#
76900	ORG+IZ
77070	PAP.TA
77200	PATENT
77500	PERCE#
77700	PERMUT
78000	PHONEM
78300	P+GRAP
78700	PHRAS#
78950	PLAN##
79300	P+DICT
79500	PRINT#
79900	PROCES+NG
80400	P+PERT
80700	PSY
81000	PUN+T+AT#
81300	Q+R
81600	RAN.AC
81800	RAP+D.SEL
82000	RECALL
82300	REDUND
82500	RELATI
82700	REPORT
82900	RES+CH
-2	RIGHT
83500	RULY.E
83900	SCAN##
84400	S*D*I#
84600	SELF[ORG]
85000	SENTEN
85230	SIG
85300	SIMULA
85600	SMART#
85700	SOCIAL
83600	SOV
86200	STANDA
86500	STOCHA
86800	STRING
87100	SUBJEC
-2	SURE##
87500	SYL+BU
87800	SYNTAC
88100	SYNTOL
57500	TALK##
88600	TEACH#
88600	TWX
88900	TERM##
89300	THEOR#
-2	THI.SO
89600	TH+GHT
89900	TIN.ST
90200	TOX
90320	TR+IST
90505	TR.MIS
90800	TYPE[SET]
91100	UP*DAT
91300	USE
91560	VER###
91800	WALNUT
92100	WORD[PAIR,GROU]
92350	XER
1	/
0	

Figure 5 (Cont'd.) Referent Items and Term Numbers

"LIB" and is immediately followed by "CONGR" in the same word or is followed within three words by a word starting with "CONGR" is recognized as being "the LIBrary of CONGRess".

As a further example of the use of this operator, all of the following will be detected by A*C*M#:

Association for Computing Machinery
Assn. for Comput. Mach.
A. C. M.
A C M
ACM

The final operator is a proximity operator such as occurs in the referent item, LANG(ART,SYN,FORM)# which first tests a user response word for the initial characters "LANG" and then scans the three previous and the three following user words for the initial character sequences "ART" then "SYN" and finally "FORM". If any of these are detected, the word starting with "LANG" is understood to mean "ARTificial LANGuage". Any of the following would be so detected:

FORMally derived LANGuages
ARTificial LANGuage
a LANGuage that is SYNthetic
a LANGuage that is FORMalized
a SYNthetic computer generated LANGuage

Each user word is tested to see if it satisfies any of the referent items in its alphabetic subgroup, i.e. those that start with the same alphabetic character.

Each time a satisfactory match is made, the number of the index term concerned is added to a hit list. By the end of the analysis of a given user response this list represents the computer's immediate interpretation of what the user said. The numbers in the hit list are in the order that these terms were detected. The hit list is limited to sixty-two items per user response. If this number of terms is exceeded as the user makes his response, he will be interrupted and the Listen link will call Check as though the user had finished his reply. Except for the limitations of the size of the hit list, the user may use as many lines on the teletype paper as he wishes to make his response. The end of a response is indicated by a blank line.

In addition to the actual lookup procedures, Listen also includes the necessary programming for accepting the user input; for treating words that are hyphenated between lines; for organizing words in preparation for the lookup procedure; for detecting the end of a response and for detecting a special character which is used to initiate a call to the Now link for the modification of the control vector. All non-alphanumeric characters except for the hyphen and the slash are treated as blank spaces.

The Listen link performs its operations at a rate of roughly two seconds per line of input. In order to facilitate the requirements of the proximity operator, Listen lags behind the user by one to two lines in its analysis of his input.

The list of referent items is being improved with operational experience. Listen makes misinterpretation errors in about 5% of the items added to the hit list. This error rate is quite tolerable. Since detected terms are read back to the user as a part of the conversation, the user has an opportunity to clear up such misunderstandings.

The original referent items were formed from C I S index terms with the following rules of thumb in mind:

1. Always use the first character of the index term.

2. Avoid the use of vowels as a general rule as they tend

to have low information value due to their high rate of occurrence.

3. Avoid the use of double consonants (a source of misspellings).

4. Select those consonants in a term which are stressed in the pronunciation of the term.

5. Use a thesaurus to check referent items for any obvious false hits they may cause.

6. Keep items as short as possible to conserve file space.

Due to inefficiencies of the program language used and of the size of the computer's core memory, some special assembly language programs were required to fit the referent list into memory. The average referent item has seven characters (including the operators). An average of four computer words is required for each referent item.

In addition to referent items for index terms, there are also several referent items for use in detecting affirmative-negative type responses. These special referents carry negative 'term' numbers and hence are easily separable from the regular C I S index term numbers. The numbers for the special referents range in value from -1,("YES"), to -5,("NO"). Thus GRINS is capable of determining the degree of negativity of a user's response.

The hit list also contains the number of hits, including special referent items, and the total number of words tested in the user's response. Usually about twenty percent of the total user words are detected as referent items. A new hit list is established for each user response.

When the analysis of the user's response is completed the Check link is called.

D. Check Link

This link converts the detected index terms, if any, from the hit list assembled by Listen to a vector of term weights. This vector, the response vector, has one element per index term and is arranged in term number order so that the value of the n^{th} element always corresponds to the weight assigned to the n^{th} term.

Each index term appearing in the response list is added to a push-down list of terms if it does not already appear in the list. This list is used by Asker as a source of terms to include at insertion points in sentences to be used as feedback to the user. For example, if, at some point in a negotiation, the pushdown list contains the numbers the following terms in the following order:

indexing
documents
automatic

and the user responds with:

I WANT INDEXING WITH TITLES AND ABSTRACTS

the list will be lengthened by the underlined terms which are detected by Listen, (Indexing is already in the list and therefore will not be added to it again.):

abstracts
titles
indexing
documents
automatic

The GRINS reply might be:

WHY DO YOU WANT ABSTRACTS, TITLES AND INDEXING?

The pushdown list would be reduced to:

documents
automatic

The pushdown list is limited in size to sixty-two terms. When this limit is reached the list is truncated with the older half of the list being destroyed.

The weights assigned to the elements of the response vector are obtained by taking the square root of their frequency of occurrence in the user response. When a term has not been used, the weight of its corresponding element will equal zero. Thus, for the first mention of a term in the user response, it gets a weight of 1.0. With each succeeding usage the term receives a successively smaller additional weight increment.

If the hit list did not contain any index term numbers but did contain one or more negative-affirmative responses, the average number of such responses is calculated. The Check link thus determines the quality of the user response. There are four such qualities possible:

1. No usable terms detected (the hit list is empty).
2. There were no index terms detected and the average of the special referents is negative.
3. There were no index terms detected and the average of the special referents is affirmative.
4. At least one index term was detected.

The Check link uses the current conversation mode and the quality of the response obtained from the user to determine which link to call next. This decision structure is given in Figure 6.

MODE	QUALITY			
	1	2	3	4
1 to 7	Asker	Asker	Asker	Asocat
8	Asker	Asker	Dump	Asocat
9	Termin	Termin	Asker	Asocat
10	Search	Search	Asocat	Asocat

Figure 6. Decision Structure for the Next Link After the Check Link

E. Asocat Link

The Asocat link serves to integrate the response vector established by Check with a search sepcification vector. The search specification organization is identical with that of the response vector. The specification represents a mapping of the user's expressed need with respect to the coordinates of the C I S document collection.

The weights of the response vector are first normalized to a maximum value of 1.0 by dividing each weight by the highest weight in the vector.

The response vector is then added to the search specification vector. This vector sum is again normalized to a maximum value of 1.0.

The difference between the previous and current search specifications is calculated by:

$$\text{difference} = \frac{\sum_{i=1}^{n} (S_{0_i} - S_{1_i})^2}{N}$$

where: n is the number of index coordinates in the vector

N is the number of elements with a value greater than zero

S_{0_i} is an element of the previous specification

S_{1_i} is an element of the current specification.

This difference is interpreted as the degree of change, or as the distance that the user's expression of his need has shifted as a result of his latest response. It is characteristic of this measure that it generally decreases toward an asymptote with each succeeding interaction as the user's definition of his need stabilizes (Figure 7). If, however, the user leaps to a different subject area the measure will increase in value. The asymptotic level that the difference measure tends toward reflects the degree of generality of the user's expressed need. It is a measure of the size of the neighborhood of the user's interest.

The rate of change of this difference measure thus reflects the stability of the user's expression of his need (Figure 8). When the average rate of change reaches a minimum threshold value further negotiation at this (conversational) level promises little in the way of better definition of the user's need. Hence, at this point another strategy, the actual search in this case, will be adopted.

The rate of change is computed as the difference between two, two-point moving averages of the difference between the old and new search specification vectors.

If the user's expression of his need has stabilized, the Search link will be called, otherwise the negotiation will be continued by calling the Asker link.

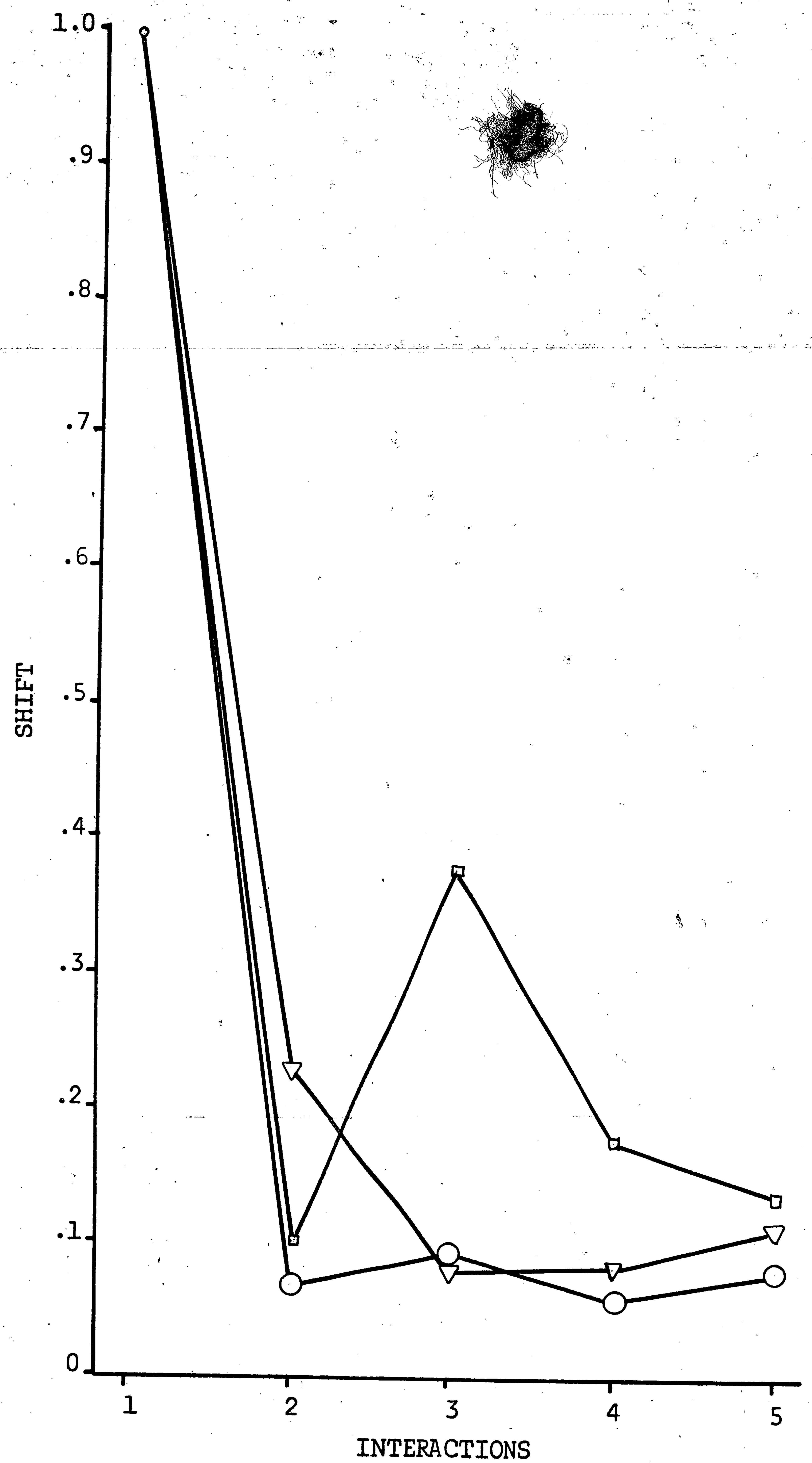

Figure 7. The Shift of the User's Expressed Interest as a Function of the Number of Interactions for Three Negotiations

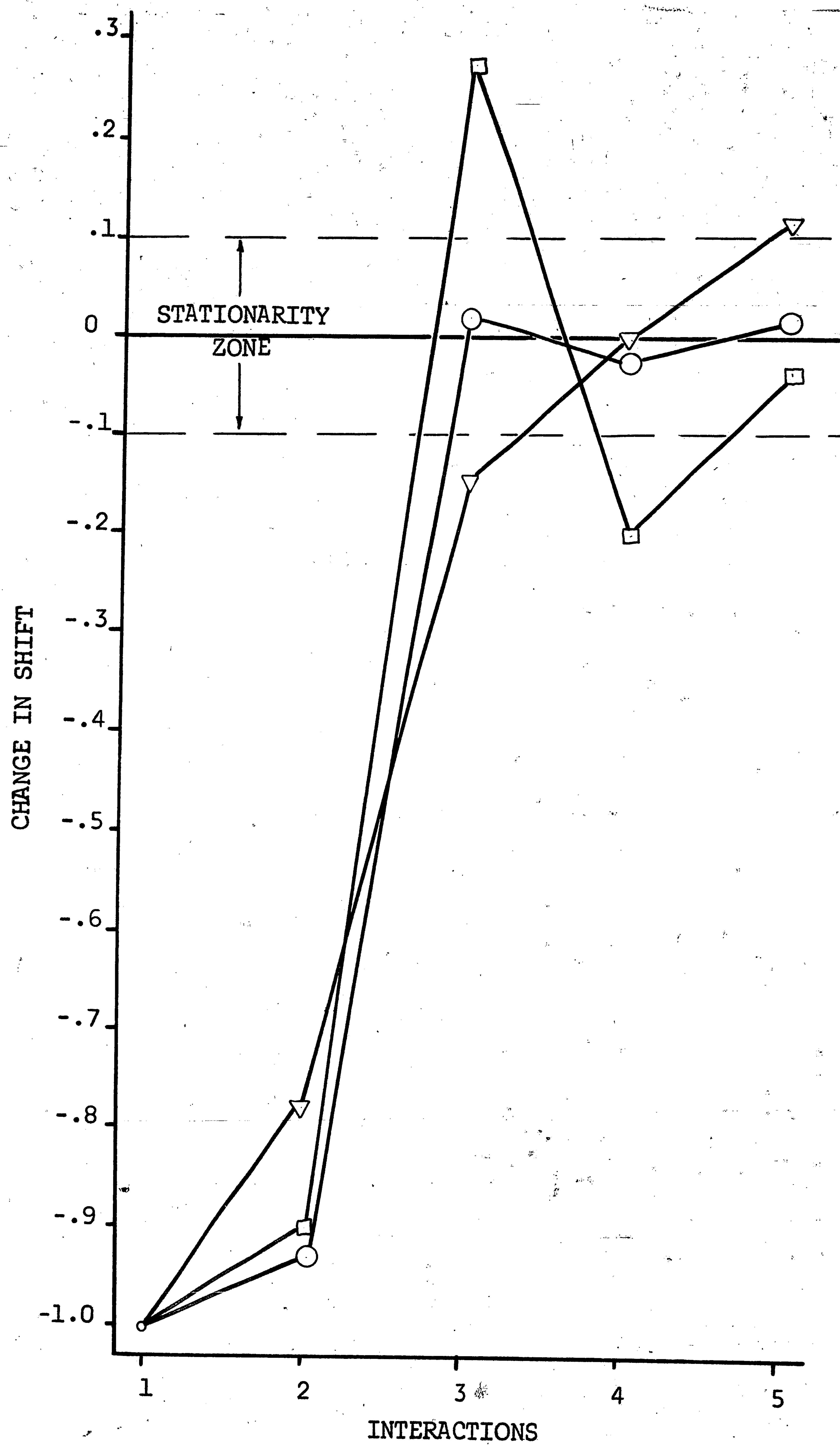

Figure 8. Change in shift of the User's Expressed Interest as a Function of the Number of Interactions for Three Negotiations

F. Asker Link

The Asker link serves to present feedback to the user. There are a hundred basic sentences which are divided into ten sets of ten each, as shown in Figure 9. Each set of sentences is designed for a separate mode of conversation. A description of the ten modes appears in Figure 10.

Many of the basic sentences may have up to three words or phrases inserted in them when they are presented to the user. For example, one such basic sentence is:

HOW IS / RELATED TO /?

The slashes are the index term or phrase insertion points, for example, if the numbers for the terms Recognition and Psychology appear at the top of the pushdown list and the above sentence were selected it would be typed out as follows:

HOW IS RECOGNITION RELATED TO PSYCHOLOGY?

The inserted words or noun phrases are obtained from the top of a push-down list of term numbers which are identified with previous responses that the user has given and which correspond to those which the Listen link has recognized as C I S index terms. This procedure permits in excess of 900 million unique sentences or questions to stimulate the user to describe his information need.

Initially the pushdown list will be empty so that none of the insertable sentences will be utilized until such time as the pushdown list contains sufficient terms to fill the sentence. As each term is used it is removed from the pushdown list. No term may appear twice at one time in the pushdown list.

```
MODE 1
WHAT DO YOU WANT
COULD YOU CLARIFY THAT
THATS NOT CLEAR TO ME, PLEASE REPHRASE IT.
PLEASE CONTINUE
NOW, TELL ME AGAIN WHAT YOU ARE LOOKING FOR.
I DONT UNDERSTAND WHAT YOU HAVE SAID, PLEASE REPEAT.
I DONT UNDERSTAND, PLEASE SAY IT MORE CLEARLY.
TELL ME MORE.
AT THIS POINT WHAT YOU ARE SAYING IS UNCLEAR TO ME. SUM UP.
WHAT ELSE

MODE 2
WHAT IS /- RELATED TO
I SEE, YOU WANT SOMETHING DEALING WITH /. AM I RIGHT
CAN YOU SAY MORE ABOUT /
WHAT ABOUT /
WHY IS / INTERESTING TO YOU
DID YOU MEAN TO SAY /  WHY
PLEASE AMPLIFY ON /.
I UNDERSTAND THE -/-, WHAT ELSE
TELL ME MORE ABOUT /.
WOULD / BE AN APPROPRIATE INTERPRETATION OF WHAT YOU WANT

MODE 3
OK, I HAVE / AND /, WHAT ELSE
HOW IS / RELATED TO /
WHAT IS THE RELATION BETWEEN / AND /
HOW ARE / AND / CONNECTED
HOW ARE / AND / ASSOCIATED
DO YOU MEAN / OR /
WHAT DOES / HAVE TO DO WITH /
YOU WANT / WITH /. RIGHT
COULD YOU EXPAND ON THE RELATION BETWEEN / AND /
IS IT CORRECT FOR ME TO ASSOCIATE / WITH /

MODE 4
WHICH OF THE FOLLOWING MEETS YOUR NEED BETTER- /, / OR /
YOU APPEAR TO WANT SOMETHING AROUND /, /, AND /. RIGHT
MAY WHAT YOU WANT BE DESCRIBED AS /, / OR /
WOULD YOU BELIEVE / OR /
MY UNDERSTANDING IS THAT YOU WANT / WITH / AND /. RIGHT
DO /, / AND / TEND TO DESCRIBE WHAT YOU WANT
YOU WANT / AND / WITH /. AM I RIGHT
WHY DID YOU INDICATE /
WHAT IS THE CONNECTION BETWEEN / AND /
GIVEN A CHOICE BETWEEN /, / AND /, WHICH DO YOU PREFER

MODE 5
IS / MORE APPROPRIATE THAN /
WHAT DO YOU WANT / FOR
WHY / AND /
I CAN NOT SEE WHAT / HAS TO DO WITH /. PLEASE EXPLAIN.
WHAT DO YOU NEED / FOR
WHY DO YOU WANT /, / AND /
IN WHAT CONTEXTUAL AREA ARE YOU GOING TO APPLY / AND /
WHAT ARE YOU GOING TO USE -/- AND -/- FOR.
WHERE DO YOU WISH TO APPLY -/- AND -/-
FOR WHAT APPLICATION DO YOU WANT -/-, -/- AND -/-
```

Figure 9. Basic Asker Responses

MODE 6

ARE YOU PUTTING ME ON
DO YOU REALLY THINK THAT THAT IS IN MY COLLECTION
WHAT ELSE DO YOU WANT
WHAT ELSE DO YOU WANT BESIDES /
I KNOW THIS TAKES TIME , PLEASE BE PATIENT AND TRY AGAIN.
IM NOT ALL THAT SMART. GIVE IT TO ME AGAIN,
PLEASE REPHRASE YOUR PROBLEM.
CHEER UP THINGS COULDNT BE ALL THAT BAD. . . . OR ARE THEY
ARE YOU CONFUSED OR IS IT ME
ALL THAT YOU HAVE SAID DOESNT MAKE SENSE SO TRY AGAIN,

MODE 7

OK SO FAR
ARE THESE OK
ARE THESE WHAT YOU WANT
DO THE ABOVE SEEM TO COVER WHAT YOU WANT
DO THESE FIT YOUR NEED
ARE THESE OK
HOW ARE WE DOING
DO THESE SEEM TO BE ADEQUATE
ARE WE ON THE RIGHT TRACK
OK THUS FAR

MODE 8

DO YOU WANT MORE DOCUMENTS
WANT MORE
MORE DOCUMENTS
MORE OF THE SAME
DO YOU WISH ANY MORE
HOW ABOUT SOME MORE DOCUMENTS
SHALL I CONTINUE GIVING YOU THESE
WANT SOME MORE
WOULD YOU LIKE ANOTHER BATCH
DO YOU WANT SOME MORE

MODE 9

DO YOU WISH TO TELL ME MORE
WELL THEN, SHALL WE TALK MORE
SHALL WE CONTINUE OUR DISCUSSION
WOULD YOU LIKE TO MODIFY YOUR REQUEST
SHALL WE CONTINUE TALKING
OK THEN WOULD YOU CARE TO REPHRASE YOUR REQUEST
THEN WOULD YOU LIKE TO SAY MORE
THEN WOULD YOU LIKE TO GIVE ME A FURTHER IDEA
WHY DONT WE DISCUSS THIS SOME MORE- OK
PERHAPS YOU WOULD LIKE TO SAY SOMETHING ELSE

MODE 10

ARE YOU TRYING TO TELL ME SOMETHING
ARE YOU RESTATING YOUR PROBLEM
CAN YOU DIRECT ME WITH RESPECT TO THE ABOVE
WHERE SHALL WE GO FROM HERE
YOU CANT ANSWER WITH A SIMPLE YES OR NO, CAN YOU
IM NOT SURE THAT I UNDERSTOOD YOU BEFORE. PLEASE SAY AGAIN.
BE PATIENT. NOW, PLEASE REWRITE THAT. IT MAY HELP ME.
PLEASE DONT RUSH ME. . . . NOW TELL ME AGAIN.
THIS CONFUSES ME. PLEASE TRY TO CLEAR ME UP.
YOU ARE NOT GETTING ACCROSS TO ME VERY WELL BUT KEEP GOING.

Figure 9 (Cont'd.) Basic Asker Responses

Mode	Term Pushdown List Size	Form of Computer's Response	
1	Empty	General context free response (i.e. w/o inserts) to generate a point of departure	
2	1	One term response	To generate related terms - to provide additional dimensions of reference
3	2-4	Two term response	
4	5+	Three term response	
5	5+	Response to probe "WHY?" for background associations and to make user think	
6	5+	Mood breaking response - to get user to relax, reflect, think more broadly, etc.	

These Modes Cycle 4,5,6,4,5,6,etc. when pushdown ≥ 5

Mode	Conditions	Response
7	After a "Dump"	"DOCUMENTS OK?"
8	After Affirmative response to 7 (no term detected)	"MORE DOCUMENTS?"
9	After Negative response to Modes 1-8 (no term detected)	"MORE TALK?"
10	Term detected after Modes 7-9	"YOUR ANSWER IS LARGER THAN MY QUESTION-ARE YOU RESTATING YOUR QUESTION?"

Figure 10. Asker Response Modes

The pushdown list is structured on a first-in-last-out basis so that the most recently identified terms are used first, thus keeping the conversation as up to date as consistent with a productive conversation. The pushdown list functions in a similar manner to one described for the ELIZA system at MIT (5).

The mode of the sentence to be presented to the user is determined by Asker on the basis of the type of (a) response last obtained from the user, (b) whether that response followed a listing of documents, (c) the size of the pushdown list and (d) the previous mode. The detailed decision structure for mode selection is given in Figure 11.

Once a new mode has been selected, one of the ten sentences for the mode is selected at random (via a random number generator) and is presented to the user together with the appropriate terms corresponding to numbers obtained from the top of the pushdown list.

G. Search Link

The search is implemented by forming a "document array" which is effectively an array with an element for each document in the C I S collection. Actually, the document array is broken into segments of 704 documents each, because of the limited size of core memory. Each segment is assembled individually and stored on disc. Presently, four such segments are required but the number may easily be expanded by modifying the control vector accordingly.

The inverted file entry is obtained from disc for each term whose element in the search specification vector has a weight above the minimum threshold value. The weight assigned to the term is modified to reflect its information theoretical value with respect to the C I S

New Mode	Last Mode	Last Response Quality	No. terms in Pushdown List	Was Dump Called?
1	1-6	1,3,4	0	no
	9	3	0	no
	10	-	0	no
2	1-6	1,3,4	1	no
	9	3	1	no
	10	-	1	no
3	1-5	1,3,4	2-4	no
	6	1,3,4	2 +	no
	9	3	2-4	no
	10	-	2-4	no
4	1-3	1,3,4	5 +	no
	9	3	5 +	no
	10	-	5+	no
5	4	1,3,4	5 +	no
6	5	1,3,4	5 +	no
7	1-6	1,3,4	-	yes
	8	3	-	-
	9	3	-	yes
	10	-	-	yes
8	7	3	-	-
9	1-6	2	-	-
	7,8	1,2	-	-
10	7,9	4	-	-

Figure 11. Mode Decision Structure

collection and then added algebraically to the elements of the document array that correspond to documents indexed by the term. In this manner weights are accumulated for each document in the collection in proportion to the extent that they correspond to the search specification vector.

The sixty-three documents with the highest weights are then assembled into a document output list in rank order of their accumulated weights. This is accomplished by means of a string processor. A typical distribution of these weights is given in Figure 12.

The search specification vector is then set to zero in preparation for any later renegotiation.

The ordered document output list of the sixty-three best documents is then stored for access by the Dump link. The document search is then complete.

The Dump link is then called.

H. Dump Link

The Dump link is first called by the Search link in order to print out the rank-order, document number, accumulated document weight, author, title, bibliographic information and location of the highest four documents on the document output list created by the Search link, which the user has not yet had presented to him during this encounter with GRINS. In the case where a search is made on the renegotiation of a question, a new search will often result in some documents that the user has already had presented to him. In this case only the rank-order, document number, and document weight together with a "SEE ABOVE" message are presented in order to save time.

When the Dump link is called the second or subsequent time after a

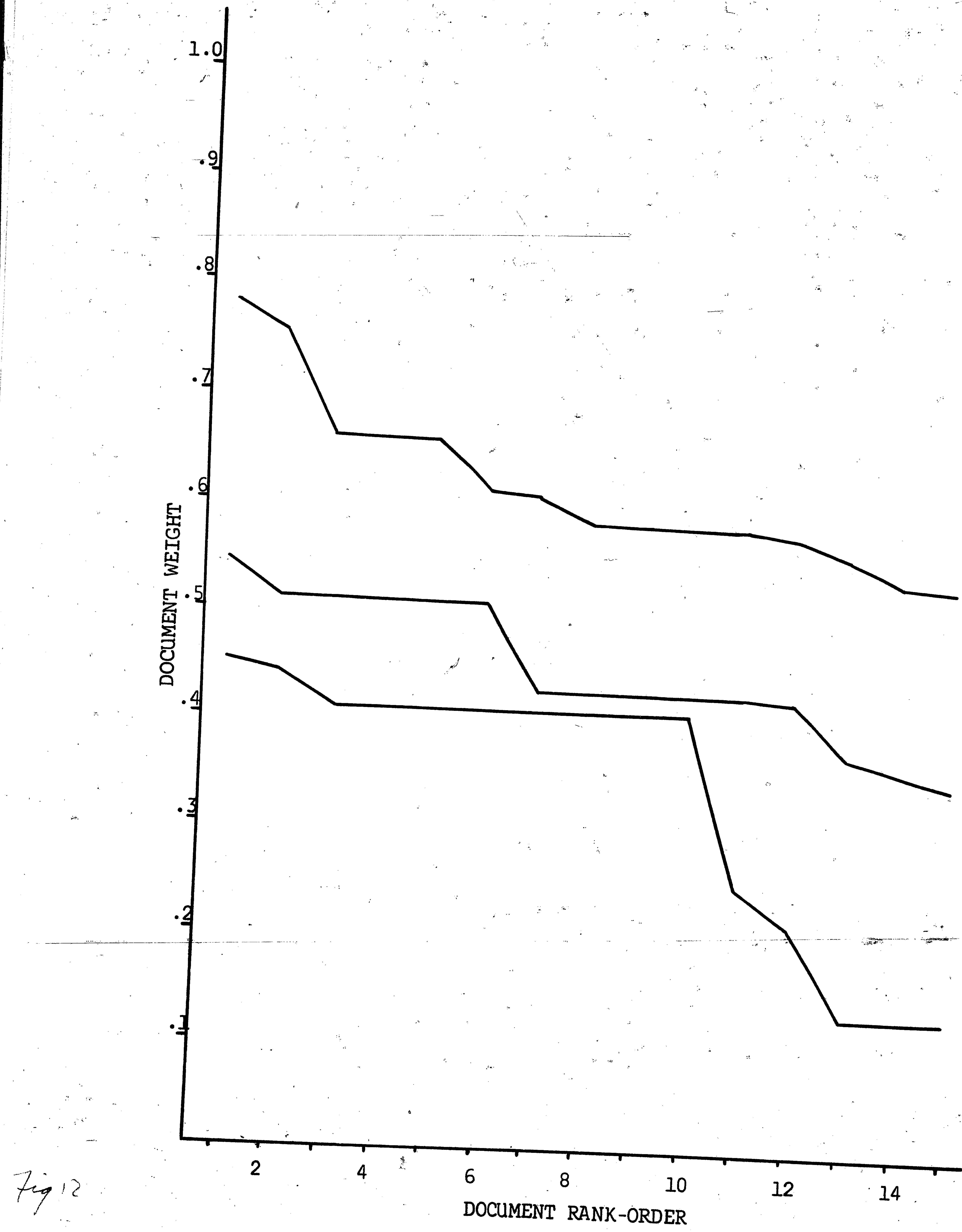

Figure 12. Distribution of Document Weights for Three Searches of the C I S Collection

search, four more documents are obtained from succeeding positions of the output list and are presented to the user in the same manner.

In the case where the output list of documents is exhausted, the user is so informed and GRINS reverts back to the conversation modes for further negotiation.

The Asker link is called after four additional new documents have been listed (in addition to any number of "SEE ABOVE" entries). The user is first asked (mode 7) if the documents appear to be what the user wants and then, if so, does the user wish more (mode 8). If not, the user is invited to renegotiate the question (mode 9). If the user declines this invitation, the Termin link is called. The foregoing sequences are illustrated in block diagram form in Figure 13 and in the sample copy of a negotiation in the Appendix.

I. Termin Link

This link serves to terminate or suspend interaction with the current user and to initiate contact with another user or to end the program.

The current user is asked if his name should be saved. If the answer is yes, the user's name and up to fifteen term numbers from the pushdown list are stored in an appropriate section of the user file on disc.

The user is then asked if there is anyone else who wishes to use the system. If there is, GRINS says to put him on, says good-by to the present user, and reminds him to log his call. Then the Init link is called.

If no more users are waiting, the Termin link copies the user file from disc and a control vector from core memory to magnetic tape. All tapes are rewound and GRINS is terminated.

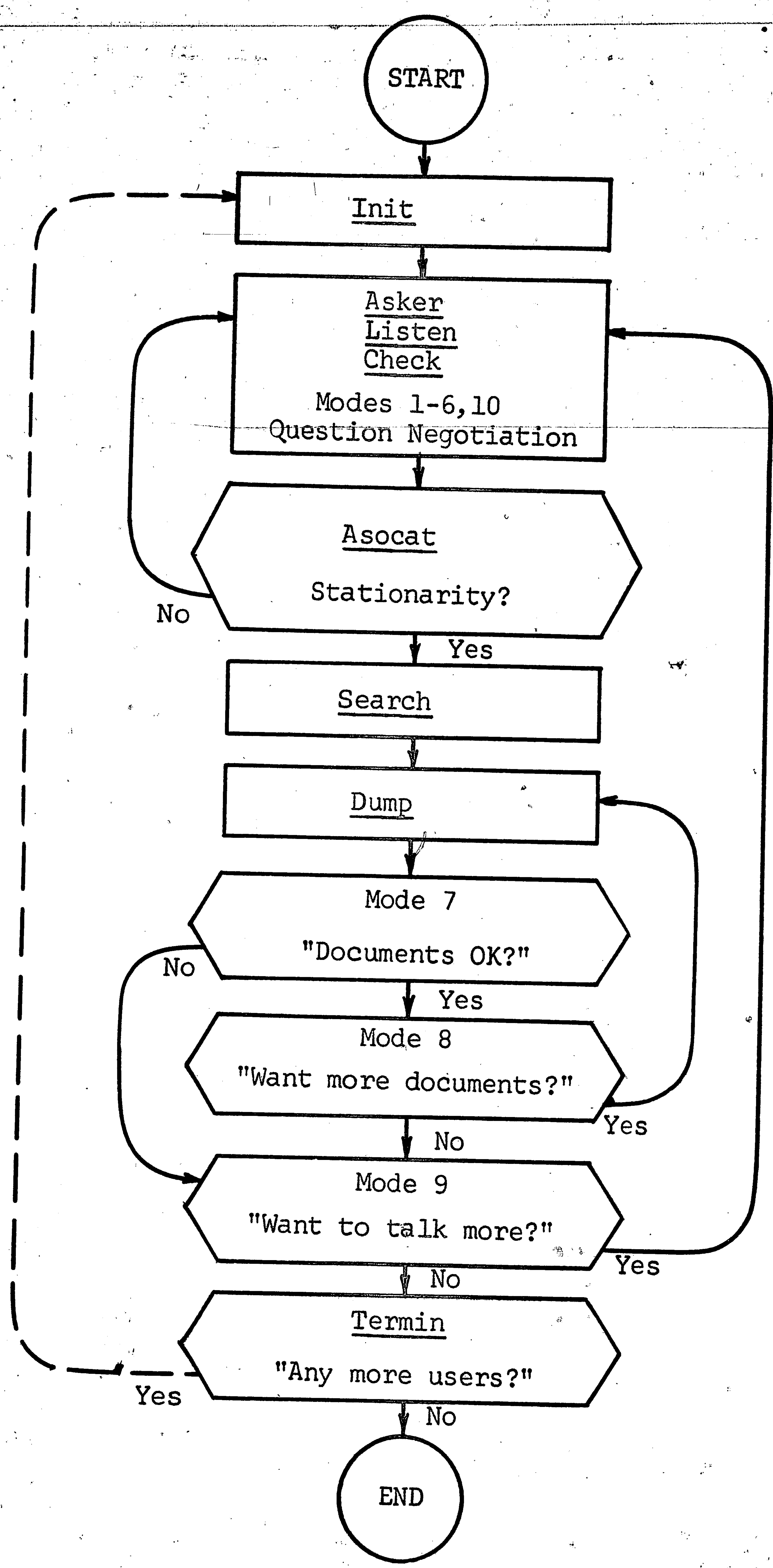

Fig. 13

Figure 13. Simplified Diagram of Negotiation Sequence

J. Now Link

This link is not used in the normal operation of GRINS. Its purpose is to offer a means of observing and modifying elements of the control vector.

This vector consists of twenty-six elements stored at the top of the common area of core memory. These elements contain thresholds, a random number generator starter, array sizes, and the like. The entire control vector is the only array that remains intact throughout all the links of GRINS and hence is often used to store arguments from one link to another.

The Now link asks the experimenter which control element he desires to see. Upon being specified, the value of the element is printed in decimal integer, octal integer, and alphabetic formats.

The experimenter then is given the opportunity to modify this element, see another element or to call in another link.

One of the control elements contains the usual link that follows Now. The experimenter may elect to go to this link or he may designate any other link.

GRINS will then continue from whatever link is selected.

IV. Evaluation and Comments

GRINS has not had sufficient use on the basis of which to make an exhaustive analysis. However, the structure as described in Chapter III has certain features that warrant comment.

Any analysis of GRINS' effectiveness must take its dynamic behavior into account. The difference between a command type retrieval system such as those examined in the Cranfield project (6) and GRINS is like the difference between a ballistic and a guided missile. In the first case a rather intense amount of effort must be concentrated into aiming the missile before the shoot, while in the second case less intense but adequate inflight corrections may be made to insure a hit.

GRINS meets the criteria specified in Chapter II. A user can satisfactorily negotiate a document search in his conversational idiom with a minimum of instruction or format restriction. The resulting searches appear to be appropriate to the requests.

From the user's point of view GRINS is far more convenient than any of the existing C I S retrieval procedures. Because of the time required to negotiate the user's question and the type of search procedure employed in GRINS, the program is slower than the C I S 'Boolean' disc search program which starts to list documents almost immediately after a search command is made.

The rank-ordered output of GRINS is generally preferable to the accession-number-ordered listing obtained by present methods because the user has more flexibility in determining the size of the listing he has requested. Within whatever size listing the user does get, the listing gives him those documents first which are determined to be most appropriate.

The GRINS search makes discriminations among documents obtained in a search whereas the 'Boolean' search does not.

Using the present version of GRINS it is not possible to make a search on the basis of the date of publication or author names. In the future a special referent could be structured to detect a date in the user's response by looking for a four character "word" starting with 19??. When such a word is detected its proximity may be scanned for such terms as "before" or "since". If such temporal modifiers are detected a special search subroutine could be utilized to limit the search accordingly.

The problem of searching on the basis of an author's name is essentially a problem of determining that such a request is being made by the user. Since it is undesirable to make the user conform to a restrictive format, special procedures would be necessary to detect forms such as "written by _____" and the like.

Both the prime author's last name and the date of publication are format controlled and hence searchable in the C I S collection.

The Listen link in its current form cannot detect a negative request such as occurs in the user sentence:

I WANT SOMETHING ON AUTOMATA BUT NOT COMPUTERS.

This stands as a major weakness in the ability of the present system to undertake a meaningful dialogue. This will be particularly true where the Listen link makes a false "hit". When such a false hit is fed back to the user, he, the user, should be able to make a direct, negative, correction rather than let the term in error diminish by attrition over succeeding interactions.

This capability should be attainable through a redesign of Listen. There are many ways of negating an idea, or the symbol (term) that denotes it, in the natural conversational idiom. These means will have to be examined to see if certain word patterns characteristically occur in them. If such word patterns can be found and if they dependably denote a rejection of terms which fall consistently in the same syntactic locations, then a means to detect such patterns may be incorporated into Listen.

If the negative request can be identified, the remainder of the system can adequately handle negative weights in the response and search specification vectors as well as in the document array.

Since the current analysis of the user's response is not too sophisticated it is possible for an inquirer experienced in the system to adopt a sort of teletype shorthand which is compatible with the referent items. Hence, for those with experience, the effort required to communicate with GRINS may be reduced.

GRINS (or any other system) will only function as well as the quality of indexing of the collection to which it is applied permits. The C I S collection is currently manually indexed by graduate students and suffers by all that is implied by manual indexing. Different indexers will index the same document differently and the same indexer will index the same document differently at different times. As an example of this, one document was discovered to be mistakenly included in the C I S collection twice. The document was indexed twice by the same person within a few weeks, each time eight index terms were used. Of these eight terms only four were used twice (and these four appeared in the title of the document). It is felt that many terms are neglected because the indexer just does not happen to consider them.

Until such a time that some economical and reliable form of automatic indexing is adopted, the system will require manual indexing.

The Listen link, with or without an added associativity feature, can be used as an aid to manual indexing. The procedure here would require that new document titles be first introduced into the serial file without indexing, then the Listen link (or a modified version of it) would be used to scan the titles of these new documents.

All terms in the titles that correspond to index terms would be listed for the indexer's consideration. If associativity is used, then associated terms can also be presented to the indexer. The indexer then may choose among these terms and add more as he sees fit.

Likewise, when new terms are added to the inverted file, the existing document listings may be scanned by Listen in order to locate old documents that might be additionally indexed by the new term.

An associativity technique has been designed, although as yet it has not been implemented. With this technique additional terms are added to those detected in the user response.

The weights of the additional terms are computed on the basis of the weight assigned to the user's term in the response vector and on the proportion of times that the associated term co-occurs with the user's term:

$$A_j = R_i \times \frac{T_{ij}}{T_i}$$

where: R_i is the weight of the i^{th} element of the response vector

T_{ij} is the frequency of co-occurrence of terms i and j

T_i is the frequency of occurrence of term i

A_j is the weight assigned to associated term j

Roughly five seconds are required to form each file entry. Hence, it would be advantageous to create and maintain a pre-computed file of associations.

In actual practice an associativity file will be stored on disc with a record for each term. The numbers of the most highly associated terms and their respective associativity factors will be stored in this record.

The additional time estimated for the associativity, using pre-computed associations will be on the order of 0.2 seconds per term. The time required for a search will increase by a factor of four due to the additional terms generated unless higher search thresholds are used.

As was mentioned previously, GRINS has been applied to a collection of twenty-six hundred documents. The time required to search the collection is increased by about four additional seconds per additional term with a weight above the threshold used in the search specification due to the fact that each term record must be obtained once for each segment of the document array. The use of a higher threshold will, therefore, shorten the search time. It may be useful in this respect to develop an algorithm to compute an optimal threshold on the basis of the distribution of weights in the search specifications. Since the search specification is normalized such an adjustable threshold may not be worthwhile.

The search speed may obviously be increased by the use of a more rapid computer installation, by more efficient programming and most dramatically by an increased size of core memory. If a 16K core memory were available for the existing program for example, each segment could be increased to handle over forty-eight hundred documents and the search rate would be increased by a factor of six or eight.

An additional alternative, which may prove necessary for very large collections (in excess of a hundred thousand documents) would be to introduce an intermediate level of searching between the conversation and the actual search levels. This intermediate level might be structured as a pseudo-document level in which statistically derived document clusters are examined prior to the real search of the documents, in order to search only those documents that appear in the more fruitful clusters.

The current version of GRINS is expensive to operate. A typical negotiation requires twenty or more minutes at a current computer time charge of $50 per hour. About eighty per cent of this time is taken by the user in making his responses.

Any new system must prove its economic worth if it is to be accepted as a real working tool. In this sense an interface system such as GRINS must be economically acceptable to both the individual user (whose economics concern time, effort, or expense) and to the administrator who must show someone that his money is being well spent.

The GRINS structure is only economically justified where a sufficiently large document collection, user population, and time-sharing computer facility are available.

Assuming that an economically satisfactory environment is established for GRINS, the structure may be extended to include a current awareness or S.D.I. capability for the known user population. A procedure for determining a user's interest with respect to the scope of the collection should be simple to devise. Such a capability could be added as an extension of the <u>Found</u> link.

Appendix

Two actual encounters with GRINS are demonstrated here. The first was conducted by the author, the second by a graduate student in the Business College at Lehigh. This student's name has been changed here to "Smith" in order to preseve his anonymity.

The lines which start with a question mark contain user responses. All the remaining lines are generated by GRINS.

The user file used for this demonstration was new so that neither the name "Green" nor the name "Smith" were familiar to the system.

Initially, after determining Green's name, GRINS loads files from magnetic tape (#IR-4, created on Sept. 6, 1967) to the disc unit.

Following this, GRINS attempts to locate Green's name in its files. Since the name isn't there, GRINS checks to see if it should be by asking, "HAVE WE MET BEFORE?" If Green had answered "YES" to the query, the system would ask for his name again and recheck the files.

At this point the actual negotiation starts. Notice that, as the negotiation develops, more and more information is obtained by GRINS. By the fifth interaction the user is largely repeating himself. His definition of his need has stabilized and so a search of the C I S document collection is made.

From the best four documents obtained in this search, it appears that the system has misinterpreted Green's use of the word <u>negotiation</u> (which is translated directly into the index term <u>conference</u> by the system). In the C I S collection the term <u>conference</u> is used to refer to mean a meeting of a professional group rather than a mediation. The second, third and fourth documents listed are inappropriate to Green's needs as a result of this error.

Green renegotiated his question upon determining the misinterpretations made by the system. The result of this second negotiation is shown on the succeeding page.

Although Green was satisfied with the new search, he did not request additional documents and he did not wish to renegotiate his need so his interaction with GRINS was terminated.

Since an additional user was indicated before Green left GRINS, the system reinitialized itself in order to greet him. This time, since the files are already stored in their proper place, the system proceeds directly to identify the new user, Smith.

It is emphasized that Smith's interests do not overlap the contents of the C I S collection very much. His example is used here as a sort of 'worst case' to show that GRINS can work even with a novice who wants something that is not represented in the document collection.

The only instructions given to Smith were to repeat grossly misspelled words, to hyphenate words when necessary at the end of a line, how to end a line, how to end a response, and to wait at the start of each line until the question mark is printed.

The difficulty that a novice can have is evident from the extensive mistypings that appear in Smith's responses. The structure of GRINS is such that such errors can be tolerated.

With the exception of the first document, all of the documents retrieved and presented to Mr. Smith are indexed by the term <u>business</u>. The second document was also indexed by <u>system</u>. The first document retrieved for Smith represents an error caused by the detection of <u>ACM</u> in his last response before the search:

NO. LET ME TRY <u>A</u>GAIN. <u>C</u>AN YOU TELL <u>M</u>E ...

The term ACM is currently used only once in the system, hence, its information value is maximal, and in this example, sufficient to put this document at the top of the list. This error exemplifies the need for very careful design of referent list items.

In this example information that is not represented in the C I S collection is requested. In this case GRINS presents the closest documents available.

```
SEP 06 67
HELLO, THIS IS GRINS.
WHAT IS YOUR NAME?
?J S GREEN
?
PLEASE WAIT A FEW MINUTES FOR ME TO GET READY.
MEMORY LOADED FROM SEP 06 67  TAPE IR-4
YOUR NAME IS NOT FAMILIAR TO ME MR. GREEN.
HAVE WE MET BEFORE?
?NO
?
WHAT DO YOU WANT MR. GREEN?
?I WOULD LIKE SOME INFORMATION ABOUT MAN MACHINE INTERACTION
?
DO YOU MEAN MACHINE OR MAN-MACHINE?
?I MEAN MACHINES THAT INTERFACE WITH MEN.  I AM CONCERNED WITH
?THE NEGOTIATION OF USER QUESTIONS BY REFERENCE LIBRARIANS AND
?SIMULATIONS OF THIS PROCEDURE BY COMPUTERS
?
DO COMPUTER, SIMULATION AND LIBRARY TEND TO DESCRIBE WHAT
YOU WANT?
?YES, BUT I AM CONCERNED WITH THEIR EFFECTS ON THE USER OF THEM.
?
WHY DO YOU WANT EFFICIENCY, REFERENCE AND QUESTIONS?
?YOU ASSUME THE FORMER, BUT YOU ARE CORRECT.  I AM BUILDING A
?DEVICE WHICH I HOPE WILL BE ABLE TO SIMULATE THE NEGOTIIATION
?PROCESS BETWEEN USERS AND INFORMATION SPECIALISTS.
?
PLEASE REPHRASE YOUR PROBLEM.
?GIVE ME REFERENCES TO WORK DONE IN THE AREA OF QUESTION NEGOTIA-
?TION PROCEDURES, AGUMENTATION AND OR SIMULATION OF THESE AND
?RELATED MATERIALS
?

A SEARCH IS NOW BEING MADE ON THE CIS DOCUMENT COLLECTION TO
LOCATE DOCUMENTS THAT BEST FIT YOUR NEED.


THE DOCUMENTS CLOSEST TO YOUR REQUEST ARE:
  1 CIS DOC NO. 1966, WT.= 0.541
HIEBER,C.  AN ANALYSIS OF QUESTIONS AND ANSWERS IN LIBRARIES
  LEHIGH UNIV CENTER FOR THE INFORMATION SCIENCES,JUNE 1966.AF-AFOSR-
  724-66, OF CRP S-235. STUDIES IN THE MAN-SYSTEM INTERFACE IN LIBRARIE
  S,REP.1
  (FILE)

  2 CIS DOC NO. 1423, WT.= 0.514
AMERICAN PSYCH ASSN.  PROJ ON SCI INFO EXCHANGE IN PSYCH
  AMER PSYCHOL ASS, PROJ ON SCIEN INFORM EXCH IN PSYCHOL, 8 VOLS,
  AUG 1963
  (FILE)

  3 CIS DOC NO. 1119, WT.= 0.514
BAUM, H.  DOCUMENTATION OF TECH. AND SCIENT. MEETINGS
  AMER DOC INST, ANNUAL MEETING, 1964, PROC., 1, 243-246
  (BOOK)

  4 CIS DOC NO. 1116, WT.= 0.514
GRIFFITH, B. GARVEY, W.  EFFECT OF CONVENTION PRESENTATION ON INFO
  EXCHANGE BEHAVIOR AND RES
  AMER DOC INST, ANNUAL MEETING, 1964, PROC., 1, 201-213
  (BOOK)

HOW ARE WE DOING?
?NOT AS WELL AS I HAD HOPED. I AM MORE INTERESTED IN THE MAN
?MACHINE INTERFACE AT THE MOMENT
?
```

Figure 14. GRINS Negotiations

```
CAN YOU DIRECT ME WITH RESPECT TO THE ABOVE?
?I AM TRYING TO DO JUST THAT.  DO YOU HAVE MATERIALS, DOCUMENTS
?AND SO FORTH ABOUT THE MECHANIZATION OF THE MAN MACHINE INTERFACE?
?
 DO DOCUMENT, MAN-MACHINE AND MATERIALS TEND TO DESCRIBE
WHAT YOU WANT?
?YES
?
I CAN NOT SEE WHAT AUTOMATION HAS TO DO WITH QUESTIONS.
PLEASE EXPLAIN.
?I AM TRYING TO ASK QUESTIONS AND TO GET ANSWERS VIA COMPUTERS
?
I KNOW THIS TAKES TIME , PLEASE BE PATIENT AND TRY AGAIN.
?OK. GIVE ME WHAT YOU HAVE ON ANSWERING INFORMATION QUIRES BY
?MEANS OF AN ON LINE COMPUTER SYSTEM
?
YOU WANT SYSTEMS WITH ON-LINE. RIGHT?
?NOT  QUITE. I WANT ON LINE DOCUMENT RETRIEVAL SYSTEMS.
?
 A SEARCH IS NOW BEING MADE ON THE CIS DOCUMENT COLLECTION TO
LOCATE DOCUMENTS THAT BEST FIT YOUR NEED.


THE DOCUMENTS CLOSEST TO YOUR REQUEST ARE:
  1 CIS DOC NO. 2168, WT.= 0.791
LICKLIDER, J.  MAN-COMPUTER INTERACTION IN INFORMATION SYSTEMS
   IN M. RUBINOFF, ED., TOWARD A NATIONAL INFORMATION SYSTEM. SECOND A
   NN NATL COLL ON INFORM RETR, APRIL 1965, SPARTAN BOOKS, 63-75
   (BOOK)

  2 CIS DOC NO. 2041, WT.= 0.765
IVIE,E. SEARCH PROCEDURES BASED ON MEASURES OF RELATEDNESS BETWEEN
   DOCUMENTS
   M.I.T.PROJECT MAC,JUNE 1966.MAC-TR-29-THESIS-,NONR-4102-01.PH.D
   THESIS
   (FILE)

  3 CIS DOC NO. 2040, WT.= 0.663
EDWARDS, D.  OCAS-ON-LINE CRYPTANALYTIC AID SYSTEM
   MIT, PROJECT MAC, MAC-TR-27, NONR-4102-01 MAY, 1966
   (FILE)

  4 CIS DOC NO. 1794, WT.= 0.663
DENNIS, J. GLASER, E.  THE STRUCTURE OF ON-LINE INFORMATION PROCESSING
   SYSTEMS
   CONGRESS ON THE INFORM SYSTEM SCIEN, 2ND PROC. WASHINGTON, SPARTAN,
   5-14
   (BOOK)

ARE THESE WHAT YOU WANT?
?YES
?
SHALL I CONTINUE GIVING YOU THESE?
?NO THANKYOU
?
PERHAPS YOU WOULD LIKE TO SAY SOMETHING ELSE?
?NOT NOW
?
SHALL I REMEMBER YOUR NAME MR. GREEN?
?YES
?
DOES ANYONE ELSE WISH TO TALK WITH ME?
?YES
?
PUT HIM ON THEN.  THANKYOU AND GOODBY MR. GREEN.
PLEASE LOG YOUR CALL
    1
```

Figure 14 (Cont'd.) GRINS Negotiations

```
SEP 06 67
HELLO, THIS IS GRINS.
WHAT IS YOUR NAME?
?CH SMITH
?
YOUR NAME IS NOT FAMILIAR TO ME MR. SMITH.
HAVE WE MET BEFORE?
?NO
?
WHAT DO YOU WANT MR. SMITH?
?I AM CONCERNED WITH THE GENERAL PROBLEM OF ECONOMIC FORECASTING.
?SPECIFICALLY, I WISH TO DEVELOP A FORECASTING SYSTEM WHICH WILLC
?WILL BE MORE ACCURATE THAN PRESENT ECONOMETRIC SYSTEMS.
?
 COULD YOU EXPAND ON THE RELATION BETWEEN SYSTEMS AND ECONOMICS?

?IT IS POSSIBLE TO LOOK AS AN ECONOMY AS A KIND OF ORGANIC OR
?BIOLOGICAL SYSTEM.  NOW, IS IT POSSIBLE TO USE MATTEMATICAL
?MATHEMATICAL SYSTMES SYSTEMS TO FORECAST ECONOMIC, SPECID
?SPECIFICALLY BUSINESS, CONDITIONS.
?
YOU WANT BUSINESS AND ECONOMICS WITH MATHEMATICS. AM I RIGHT?
?WELL.... NOT EXACTLY. I WANT TO USE MATHEMATICS TO FORECAST
?FUTURE BUSINESS OR ECONOMIC CONDITIONS.
?
 I CAN NOT SEE WHAT BUSINESS HAS TO DO WITH USE. PLEASE EXPLAIN.
?BUSINESS HAS NOTHING TO DO WITH USE, RATHER I WISH TO USE
?THE LANGRAG  LANGUAGE OF MATHEMATICS TO FORECS  FORECAST BUSIN-
?ESS FUTURE BUSINESS CONDITIONS SUCH AS THOSE WHICH EXIST IN
?THE HOUSING MARKET.
?
 CHEER UP THINGS COULDNT BE ALL THAT BAD. . . . OR ARE THEY
?NO.  LET ME TRY AGAIN.  CAN YOU TELL ME ANYTHING ABOUT
?BUSINESS FORECASTING.
?
 A SEARCH IS NOW BEING MADE ON THE CIS DOCUMENT COLLECTION TO
LOCATE DOCUMENTS THAT BEST FIT YOUR NEED.


THE DOCUMENTS CLOSEST TO YOUR REQUEST ARE:
  1 CIS DOC NO. 1648, WT.= 0.455
RUBINOFF, M.  DESCRIPTION OF CATALOGING + INDEXING SYST FOR THE ACM
  REPOSITORY
  UNIV OF PENN, MOORE SCH OF ELEC ENG, AFOSR 65-0424, AD 612277, JAN
  1965
  (FILE)

  2 CIS DOC NO. 1522, WT.= 0.451
AUERBACH CORP.  A DESCRIPTION OF AUERBACH INFORMATION MANAGEMENT SYSTEM
  AUERBACH CORP, PHILADELPHIA, PA, FEB 1966
  (FILE)

  3 CIS DOC NO. 2028, WT.= 0.417
GENTLE,E.C.JR.  DATA COMMUNICATIONS IN BUSINESS
  NEW YORK, AMERICAN TELEPHONE AND TELEGRAPH CO.
  (BOOK)

  4 CIS DOC NO. 1479, WT.= 0.417
AMERICAN CHEM. SOC.  SYMP ON INFO PROBLEMS IN CHEMICAL MARKETING
  J OF CHEM DOC 5, 63-86, MAY 1965
  (L. U. LIBRARY)

DO THESE SEEM TO BE ADEQUATE?
?OK
?
```

Figure 14 (Cont'd.) GRINS Negotiations

```
MORE DOCUMENTS?
?YES PEL PLEASE
?
  5 CIS DOC NO. 1154, WT.= 0.417
SCHULTZ, L.  A SYSTEM FOR RETRIEVING UNTITLED BUSINESS DOCUMENTS
  AMER DOC INST, ANNUAL MEETING, 1964, PROC., 1, 453-456
  (BOOK)

  6 CIS DOC NO.  798, WT.= 0.417
GREENBERGER, M.  SUPPLEMENT TO OPS-1 MANUAL
  MIT, MAC-TR-8, ONR-NONR-4102(01), AUG 1964
  (FILE)

  7 CIS DOC NO.  405, WT.= 0.417
VEYETTE, J.  IMPACT OF INFO RETRIEVAL ON CORPORATE STRUCTURE
  DATA PROCESSING 4, JAN 1962
  (L. U. LIBRARY)

  8 CIS DOC NO.  303, WT.= 0.417
REES, T. ET AL  AUTOMATIC CORRELATION OF INFORMATION FOR PURPOSES OF
  COMMERCIAL INTELLIGENCE
  PERRY, J, ED, TOOLS FOR MACH LIT SEARCHING, NEW YORK, INTERSCIENCE,
  457-479 -010.78 P464T-
  (L. U. LIBRARY)

ARE THESE OK?
?YES
?
MORE DOCUMENTS?
?NO THANKYOU
?
SHALL WE CONTINUE TALKING?
?NO NOW
?
SHALL I REMEMBER YOUR NAME MR. SMITH?
?YES
?
DOES ANYONE ELSE WISH TO TALK WITH ME?
?NO
?
THEN I WILL STOP HERE.  GOODBY MR. SMITH.
PLEASE LOG YOUR CALL
   2
```

Figure 14. (Cont'd.) GRINS Negotiations

```
PROGRAM TERMINATED.
```

References

1. C. E. Hieber, "An Analysis of Questions and Answers in Libraries." Report No. 1, Studies in the Man-System Interface in Libraries, Center for the Information Sciences, Lehigh University, Bethlehem, Pa., (June 1966), p. 15.

2. R. S. Taylor, "Question-Negotiation and Information Seeking in Libraries," Report No. 3, Studies in the Man-System Interface in Libraries, Center for the Information Sciences, Lehigh University, Bethlehem, Pa., (July 1967), p. 14.

3. R. S. Taylor, et al., "Experimental Literature Collection and Reference Retrieval System of the Center for the Information Sciences," Report No. 2, Experimental Systems Studies, Center for the Information Sciences, Lehigh University, Bethlehem, Pa., (April 1967).

4. J. S. Green, Vocabulary Control in On-Line Computer Conversation Routines (I.S. 421 Report) Lehigh University, Bethlehem, Pa., (Unpublished, 1967), pp 1-7.

5. J. Weizenbaum, "ELIZA, A Computer Program for the Study of Natural Language Communication." ACM Communications, 9, (January 1966), pp 36-45.

6. C. W. Cleverdon, "The ASLIB Cranfield Research Project on the Comparative Efficiency of Indexing Systems," ASLIB Proc., 12, (1960), pp 412-431.

Bibliography

Brosin, H. W. "Expressed and Unexpressed Needs." in A. Kent and O. E. Taulbee (Ed.) Electronic Information Handling, Spartan Books, Washington, D. C., 1965, pp. 75-84.

Carlson, W. M. "Scientists' Requirements." in A. Kent and O. E. Taulbee (Ed.) Electronic Information Handling, Spartan Books, Washington, D. C., 1965, pp. 85-91.

Hieber, C. E. "An Analysis of Questions and Answers in Libraries." Report No. 1, Studies in the Man-System Interface in Libraries, Center for the Information Sciences, Lehigh University, Bethlehem, Pa., June 1966.

Licklider, J. C. R. Libraries of the Future. M. I. T. Press, Cambridge, Mass., 1965.

Mesarovic, M. D. "Toward a Formal Theory of Problem Solving." in M. A. Sass and W. D. Wilkinson (Ed.) Symposium on Computer Augmentation of Human Reasoning, Spartan Books, Washington, D. C., 1965, pp. 37-64.

Prywes, N. S. "Browsing in an Automated Library Through Remote Access." in M. A. Sass and W. D. Wilkinson (Ed.) Symposium on Computer Augmentation of Human Reasoning, Spartan Books, Washington, D. C., 1965, pp. 131-150.

Shure, G. H. and R. J. Meeker. Probing Behind the Human Decision. Systems Development Corporation, Santa Monica, Calif., July 1965.

Simmons, R. F. "Answering English Questions by Computer: A Survey." ACM Communication, vol. 8, January 1966, pp. 53-70.

Taylor, R. S. et al. "Systems Manual for the Experimental Literature Collection and Reference Retrieval System of the Center for the Information Sciences." Report No. 2, Experimental Retrieval Systems Studies, Center for the Information Sciences, Lehigh University, Bethlehem, Pa., April 1967.

Taylor, R. S. "Question-Negotiation and Information-Solving in Libraries." Report No. 3, Studies in the Man-System Interface in Libraries, Center for the Information Sciences, Lehigh University, Bethlehem, Pa., July 1967.

Weizenbaum, J. "ELIZA, A Computer Program for the Study of Natural Language Communication." ACM Communications, vol. 9, January 1966, pp. 36-45.

Vita

James Sproat Green V, son of Richard Fisher and Margaret Webster Green, was born in Elizabeth, New Jersey on March 3, 1937. He is married to the former Mary Lou Johnson of Vestal, New York. He was graduated from Thomas Jefferson High School in Elizabeth in June, 1955 and received a Bachelor of Arts Degree in Psychology from Lehigh University in January, 1966.

From 1955 until 1959, Mr. Green served with the U. S. Navy as a Guided Missile Fire Control Technician. He was employed by the Singer Manufacturing Company, Diehl Division in the design and development of automatic control devices from 1961 until 1963. In addition he was employed on a part-time basis in the Bio-Electric Laboratory of the Psychology Department at Lehigh before his present appointment as a Research Assistant in the Center for the Information Sciences at Lehigh University.

Mr. Green was granted twenty claims on U. S. Patent No. 3,150,364, "Anti-Collision Detection and Warning System," in September of 1964.