1990

Word hypothesis of phonetic strings using hidden Markov models

Jeffery W. Engbrecht

Rochester Institute of Technology

School of Computer Science and Technology

Word Hypothesis of Phonetic Strings Using Hidden Markov Models

by

Jeffery W. Engbrecht

A thesis, submitted to the Faculty of the School of Computer Science and Technology, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science.

Approved by : Robert T. Gayvert (Chairman) 21 May 1990

John A. Biles

Peter G. Anderson

Word Hypothesis of Phonetic Strings Using Hidden Markov Models

I Jeffery W. Engrecht Hereby grant permission to the Wallace Memorial Library of RIT to reproduce my thesis in whole or in part. Any reproduction will not be for commercial use or profit.

Date 25 May 90

ABSTRACT

This thesis investigates a stochastic modeling approach to word hypothesis of phonetic strings for a speaker independent, large vocabulary, continuous speech recognition system. The stochastic modeling technique used is Hidden Markov Modeling. Hidden Markov Models (HMM) are probabilistic modeling tools most often used to analyze complex systems.

This thesis is part of a speaker independent, large vocabulary, continuous speech understanding system under development at the Rochester Institute of Technology Research Corporation. The system is primarily data-driven and is void of complex control structures such as the blackboard approach used in many expert systems. The software modules used to implement the HMM were created in COMMON LISP on a Texas Instruments Explorer II workstation.

The HMM was initially tested on a digit lexicon and then scaled up to a U.S. Air Force cockpit lexicon. A sensitivity analysis was conducted using varying error rates. The results are discussed and a comparison with Dynamic Time Warping results is made.

ACM Keywords: Speech Recognition and Understanding
Natural Language Processing

ACM Code: I.2.7

Table of Contents

	Page
1. Introduction	
1.1 Problem Statement	1
1.2 Previous Work	3
1.3 Theoretical and Conceptual Development	3
1.3.1 An Introduction to Hidden Markov Models	4
1.3.2 HMMs in Speech Recognition	7
1.3.3 The Three Problems of HMMs	10
1.3.4 HMM types	15
2. Project Description and Methodology	15
2.1 Phoneme Confusion Data	17
2.2 Knowledge Used	18
2.3 Software Tools	19
2.3.1 Phone and Word Model Construction	20
2.3.2 Phoneme String Generator	20
2.3.3 Viterbi Search with Optional Beam Search Technique	23
2.3.4 Grammar Construction	25
2.3.5 Error Analysis	28
2.4 Hardware Tools	28
2.5 Lexicon Construction	28
2.6 Test Data Creation	29
2.7 Log probabilities	29
3. Results	31
3.1 Phase I - Baseline Digit Model	34

1. Introduction and Background

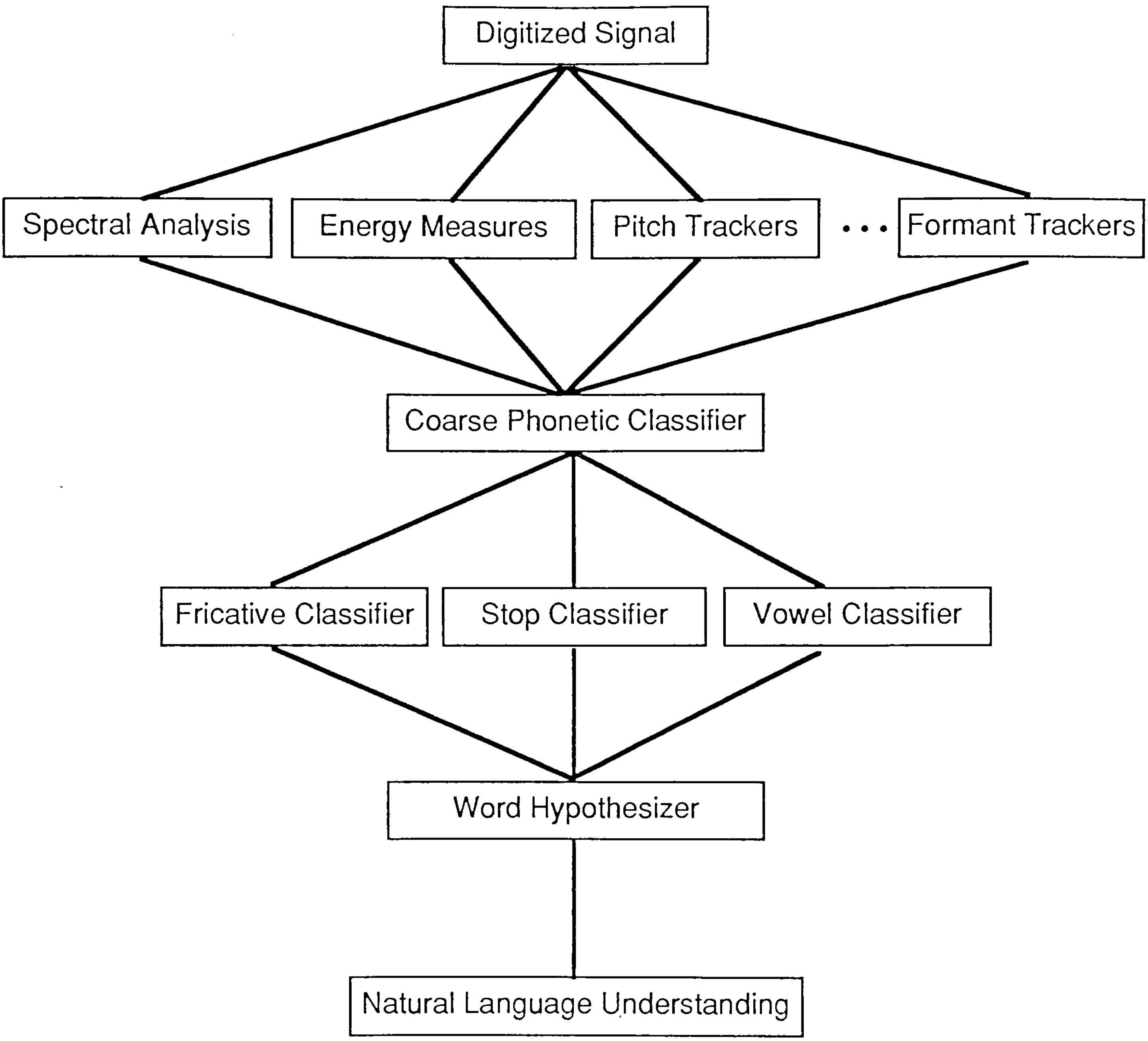
1.1 Problem Statement

This thesis investigates a stochastic modeling approach to word hypothesis of phonetic strings for a speaker independent, large vocabulary, continuous speech recognition system. The stochastic modeling technique used is Hidden Markov Modeling. Hidden Markov Models (HMM) are probabilistic modeling tools most often used to analyze complex systems. This, in addition to their inherent ability to handle time-varying processes, makes HMMs a natural candidate for use in speech recognition.

This thesis is part of a speaker independent, large vocabulary, continuous speech understanding system under development at the Rochester Institute of Technology Research Corporation. The system is primarily data-driven and is void of complex control structures such as the blackboard approach used in many expert systems. Figure 1 shows the software architecture of the project. An input utterance is processed to produce a digitized signal. The digitized signal is then provided as input to several feature extractors. The feature extractors compress the data as much as possible without losing the phonetic content. The feature frames are then sent to a knowledge based phoneme builder to produce strings of undifferentiated phonemes. Words are then hypothesized. The word hypothesis portion of the project is the focus of this thesis and will be described in detail later. The last process is the use of syntactic and semantic knowledge sources in an attempt to form a syntactically correct utterance and provide it with meaning.

The parsing of a phonetic transcription would not pose a problem if there were no errors contained within the transcription. It would be a simple matter of string comparison. However, errors introduced to the phonetic transcription increase the complexity and prevent the lexical access procedure from being a simple lexicon lookup. Front-end errors are caused by a lower level's inability to segment the signal properly and distinguish between similar sounding phonemes. A particular speech segment may include any or all of the errors of substitution, insertion and deletion. Shown below are examples of these three types of error.

Substitution Error - *battle* : b ae t el -> b ae d el



RIT's Speech Understanding System Architecture

Figure 1

Insertion Error - *chauffeur* : sh ow f er -> sh ow l f er

Deletion Error - *and* : ae n d -> n

A substitution error is one in which a similar sounding error takes the place of the proper one. An insertion error is one in which a phoneme, not in the correct phonetic transcription, is added to the string. And finally, a deletion error occurs when a phoneme is not recognized and subsequently left out. When combined, these three types of errors can present a formidable obstacle for the speech recognition system to overcome.

1.2 Previous Work

As a first attempt to solve the word hypothesis problem in the RIT Speech Project, R. Thomas Selman investigated a dynamic programming approach by using Dynamic Time Warping [SELM89]. Dynamic Time Warping (DTW) is a common method of sequence comparison used in matching the acoustic feature vectors representing an unknown input utterance and some reference utterance. DTW met with limited success. Although the accuracy obtained by using DTW was relatively good, the time required to process a phonetic string did not prove satisfactory.

In 1989, L. E. Levinson used a form of Hidden Markov Models which was independent of lexical and syntactic constraints [LEVI90]. In that study, Levinson treated word recognition as a classical string-to-string editing problem which is solved with a two-level dynamic programming algorithm that accounts for lexical and syntactic structure. Again, Levinson obtained fairly accurate results, but the time required to obtain those results is far away from our goal of real time processing.

1.3 Theoretical and Conceptual Development

This thesis is a subset of the work accomplished by Kai-Fu Lee [LEE89] on the SPHINX system and is a follow up to R. Thomas Selman's work. However, rather than working at the

phoneme recognition level, as Lee did, this thesis simulates the levels of phoneme classification. As part of the speech project, this thesis groups together given phonemes, obtained from a lower level phoneme classification routine, to represent words.

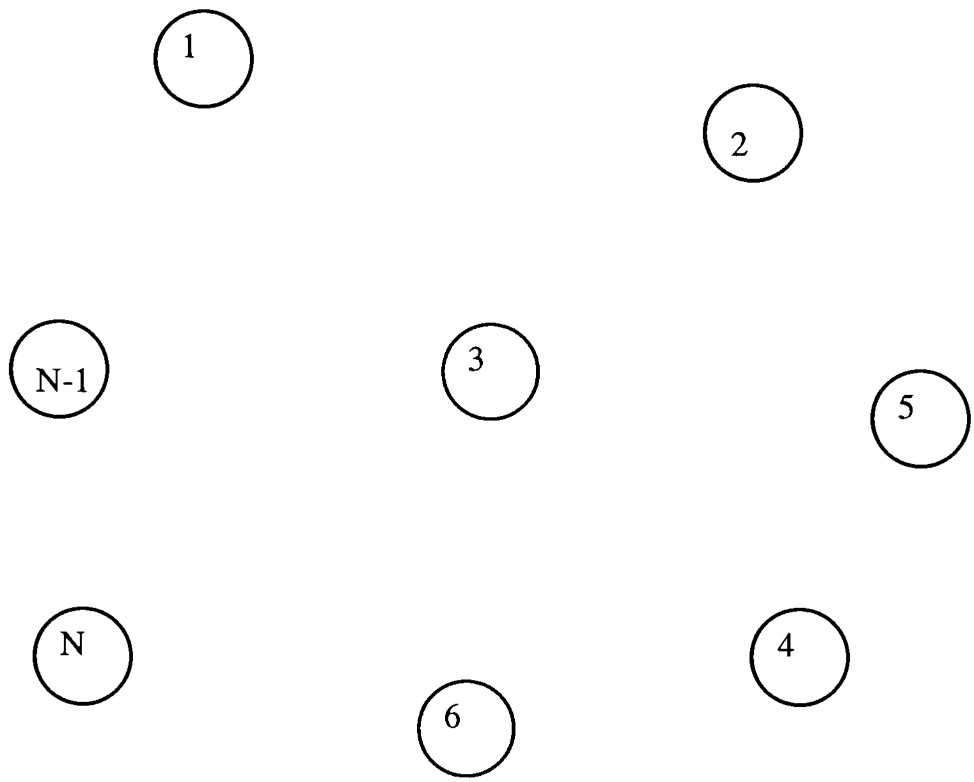
1.3.1 An Introduction to HMMs

An HMM is a doubly stochastic process with an underlying stochastic process that is not observable (it is hidden), but can only be observed through another set of stochastic processes that produces the sequence of observed symbols [RABI86]. Simply stated, an HMM is a collection of states connected by transitions. They are somewhat analogous to automata. The transitions include two sets of probabilities; an output probability density function (pdf) and a transition probability from state to state. In the modeling of speech, the pdf can define the conditional probability of an output phoneme and the transition probability can define the connection of those phonemes. The connection of multiple phonemes comprise words. Two examples are provided which provide a better understanding of how HMM's work. They are the lily pond example [HOWA71] and the coin toss example [RABI86].

Lily Pond

The lily pond example shown in Figure 2 illustrates a situation where a frog is sitting on a lily pad. The frog is permitted to jump only from pad to pad (the frog may not swim between pads). Hence, pads are represented as discrete states of the transitions possible in the lily pad. The frog may even jump and land on the same pad he is currently on. Figure 3 shows some of the possible transitions in the lily pad example. The probabilities illustrated can be represented in matrix form. This facilitates easy manipulation by computers. In a purely Markovian system, the sum of the probabilities in any row sum to 1. The transition of the frog from one pad to another is based on the Markovian assumption which states that only the last state occupied by the given process is relevant in determining its future behavior. This assumption is a very strong assumption. Very few physical systems are strictly Markovian [HOWA71]. However, certain constraints in the HMM can be relaxed to allow us to apply a semi-Markovian model to a

Lilypad - frog example



The frog must always sit on the lily pad; he never swims.

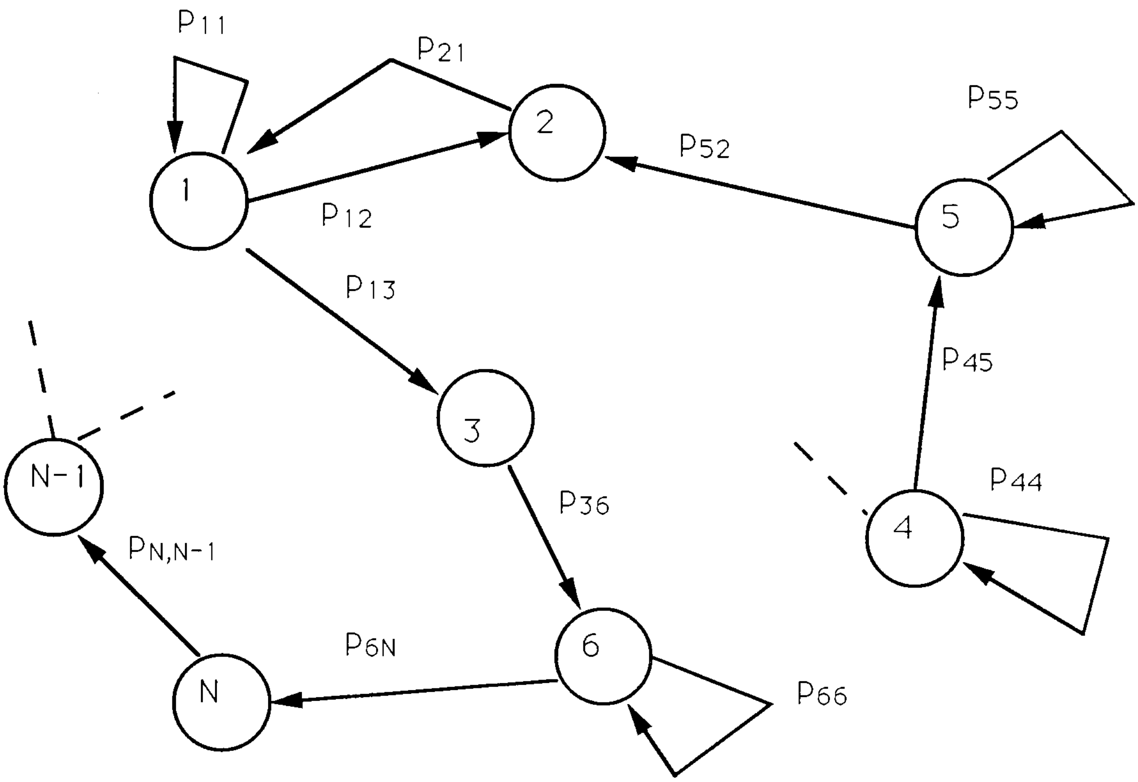
Occasionally the frog jumps in the air and lands on another pad.

Markovian Assumption:

Only the last state occupied by the process is relevant in determining its future behavior.

Figure 2

Lilypad - frog example



$$P = \{P_{ij}\} = \begin{bmatrix} P_{11} & P_{12} & \cdots\cdots & P_{1N} \\ P_{21} & P_{22} & \cdots\cdots & P_{2N} \\ . & & & \\ . & & & \\ . & & & \\ P_{N1} & P_{N2} & \cdots\cdots & P_{NN} \end{bmatrix}$$

Figure 3

wide class of systems. Chemical processes and speech recognition are but two examples of systems that can be emulated using HMMs.

Coin Toss

The coin toss example illustrates how states in a Markov model are hidden. Imagine you are in a room where there is another person behind a partition and that person is giving you results of consecutive coin tosses. The observations are described by:

$$O = o_1, o_2, \ldots\ldots\ldots\ldots o_n$$
$$h\ ,\ t\ ,\ \ldots\ldots\ldots\ldots h$$

Using HMM's to represent the hidden process occurring behind the partition can provide great assistance in determining what is transpiring to create the observations. A single fair coin is the model most people would use to describe the outcomes of heads and tails. Figure 4 shows this model along with two other models. The observed outcomes of the coin toss can also be explained by a 2 fair coins model, a 2 biased coins model, or a model with any number of coins. The probabilities associated with heads or tails vary according to the model used.

1.3.2 HMMs in Speech Recognition

Hidden Markov models are named after the Russian mathematician Andrei Markov. Markov's modeling came about quite indirectly. Markov was trying to determine if the law of large numbers applied to dependent variables as well as independent variables. In particular, Markov wondered if the sum of the dependent variables would satisfy the central limit theorem by being normally distributed. Markov published a proof verifying that it was indeed true for modulo 2 numbered variables. Markov was dissatisfied with his complex combinatorial proof and subsequently published a paper in 1907 [MARK07]. His work, although not initially

Coin Toss Example

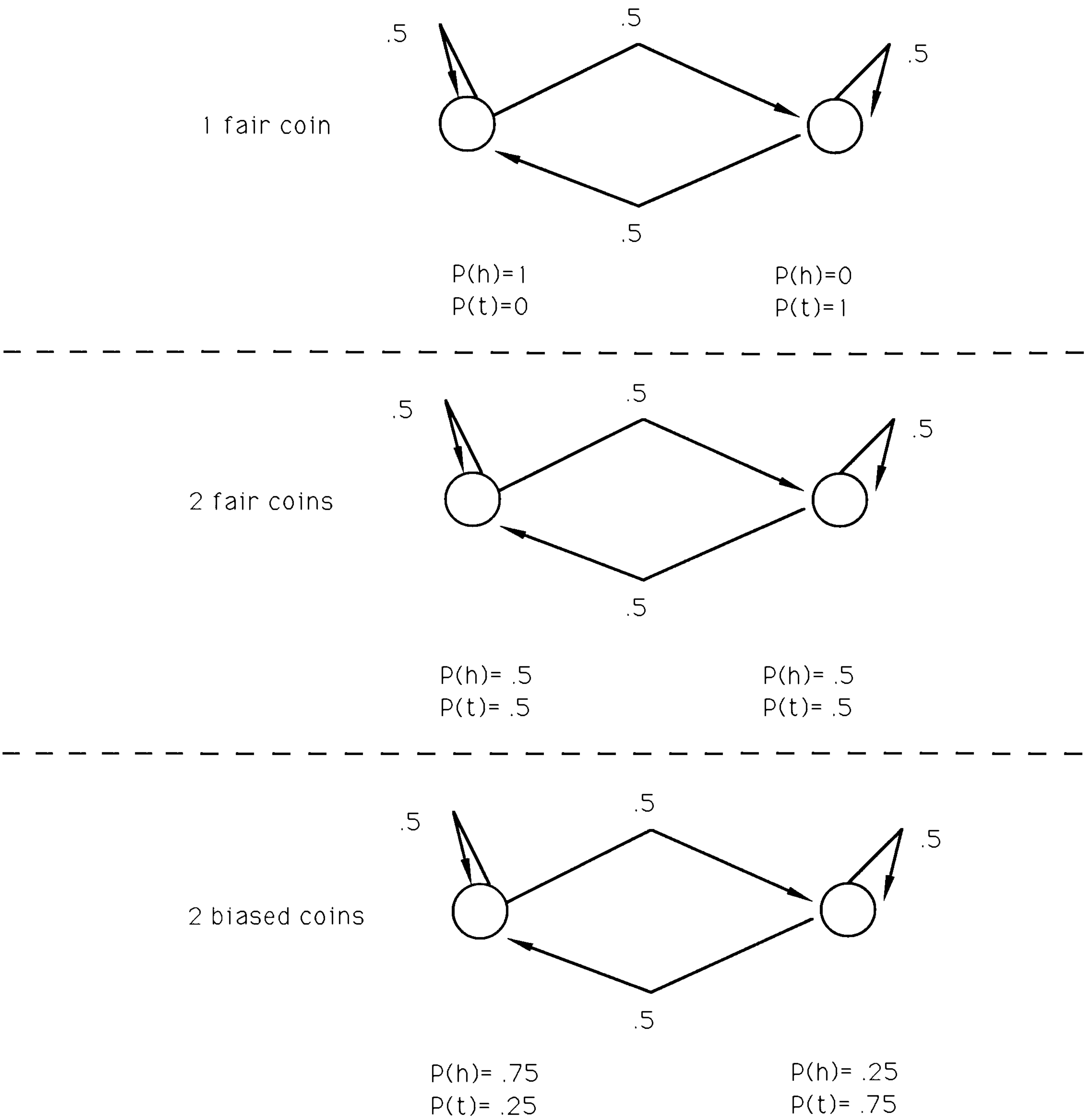


Figure 4

developed for speech recognition, formed the basis for much of the work accomplished in the speech recognition domain.

Until the early 1970's, HMMs were not used in speech recognition. This was primarily because of the tractability problem encountered while training HMMs to change their associated probabilities to more optimally model the process. A major breakthrough occurred when a maximization technique was developed in the early 1970's. This technique, known as the forward-backward re-estimation or Baum-Welch algorithm, solved the tractability problem for HMMs [BAUM72]. This breakthrough paved the way for the application of HMM's to automatic speech recognition [BAKE75], [BAKI76], and [JELI76].

Funding by the Advanced Research Project Agency (ARPA) of the Department of Defense provided much of the impetus for many of the speech recognition systems developed in the 1970's. Of the systems developed in the 1970's, the two that could be classified as forerunners of HMM's were the DRAGON system [BAKE75] and the Harpy system [LOWE77].

The DRAGON system, developed at Carnegie-Mellon University, was the first system to use a simplistic Markov technique. The DRAGON system used uniform stochastic modeling for all knowledge sources. However, the results were not as promising as were first expected. On only a 194 word speaker-dependent continuous task, DRAGON recognized 84% of the words correctly. Consequently, the interest in the DRAGON system design group dissipated and the DRAGON system evolved into the Harpy speech recognition system.

The Harpy system, also developed in the Computer Science Department at Carnegie-Mellon University, was the only speaker independent, large vocabulary, continuous speech recognition system to meet or surpass the standards set forth by ARPA. The Harpy system had a sentence accuracy of 91% across five different speakers (3 male and 2 female) and ran in less than 7 million machine instructions per second (MIPS). Although the Harpy system used state diagrams similar to those used in HMMs, CMU's approach was to combine the syntactic, lexical, and word juncture knowledge into one large state diagram designed specifically for a distinct lexical domain. That state diagram, or network, became a complete and pre-compiled representation of all possible utterances in the task language.

Since these works accomplished in the 1970's, there have been two approaches to the problems encountered in the speech recognition domain. One direction was to view speech recognition as a process which could be solved by expert systems [COLE83], [HATO84], [ADAM86], and [THOM87]. This approach so far has been less effective. The other approach

involves stochastic modeling and the use of HMMs. Waibel [WAIB86] also showed that the use of human knowledge of prosodic parameters such as stress, intensity and duration could be used to enhance performance. This approach is promising because it improves word recognition significantly [LEE89].

In 1989, R. Thomas Selman used Dynamic Time Warping in an attempt to solve the word hypothesis problem [SELM89]. His system met with limited results which will be used later for comparison.

A widely accepted theory today holds that speech recognition in humans proceeds from an intermediate representation of the acoustic signal in terms of a small number of phonetic symbols. Levinson used a speech recognition system based on this theory [LEVI90] in which the acoustic-to-phonetic mapping was done using HMMs. It resulted in a 76.6% word accuracy on the DARPA resource management task.

The latest speaker independent, large vocabulary, continuous speech recognition system developed was the SPHINX system at Carnegie-Mellon University in the late 1980's [LEE89]. The SPHINX speech recognition system used HMMs and Vector Quantization (VQ) code books for phoneme recognition. The results of the SPHINX system were promising. When the HMMs are combined with phonological rules, the system accuracy approaches 97% for word recognition using a lexicon of size 997 and a perplexity of 20.

1.3.3 The Three Problems of Hidden Markov Models

To use HMMs, a formal notation is required to address necessary variables. The formal notation used throughout the thesis is shown in Figure 5.

The variables of A, B and Π are the three most important variables in the notation used. Hence HMMs are represented as λ being a function of A, B and Π.

When using the HMM approach to model a particular domain, the three problems which are presented are the evaluation problem, the decoding problem and the learning problem.

The first problem is to determine the probability of an observed sequence ($O = o_1, o_2, \ldots, o_n$) when given a model. This involves summing the probabilities of all paths. There is a brute force approach to solve for this probability [PORI88]. This method requires (2T-1)N

Formal HMM Notation

T = length of observation sequences

N = number of states in the model

M = number of observation sequences

$Q = \{ q_1, q_2, \ldots, q_n \}$ states

$V = \{ v_1, v_2, \ldots, v_m \}$ discrete set of possible observations

$A = \{ a_{ij} \}$, a_{ij} , state transition probability distribution

$B = \{ b_j(k) \}$, $b_j(k)$, observation symbol probability distribution in state j

$\Pi = \{ \pi_i \}$, π_i , initial state distribution

$\lambda = (A, B, \Pi)$

Figure 5

multiplications and N-1 additions. The brute force method becomes quickly intractable. The forward-backward algorithm solves the first problem [BAUM72]. Figure 6 illustrates a simple HMM and the computations involved in both a forward and backward sweep of the lattice. The concept behind the forward-backward algorithm is to start at the first and last output symbols and work toward the middle of the trellis summing α and β's as you go.

The second problem is how to best choose a state sequence (I) so that it maximizes the expected number of correct individual states. The Viterbi algorithm was developed to accomplish just that [VITE67]. It works by simply determining the most likely state at every instance without regard to the lattice structure, the neighboring states in time, and the length of the observation sequence. The Viterbi algorithm can be used for segmentation, annotation, and recognition. Figure 7 highlights the formal steps in the Viterbi algorithm. The Viterbi algorithm is similar in implementation to the forward-backward calculation. However, a maximization over previous states is used in place of the summing procedure. A trellis structure efficiently implements the computation [RABI86].

The third problem involves performing a sensitivity analysis on the model. Model parameters are adjusted to maximize the probability of the observation sequence. There does not exist an analytical method to solve this problem. However, as mentioned earlier, a gradient iterative technique was developed by Baum [BAUM72]. The Baum-Welch re-estimation formulas guarantee that either 1) the probability will be improved or 2) a critical point has been reached where the probability is already maximized. The re-estimation formulas are as follows:

1. $\pi_i' = \gamma_1(i)$, $\qquad 1 <= i <= N$

2. $a_{ij}' = \Sigma\ \xi_t(i,j)\ /\ \Sigma\ \gamma_i(i)$ $\qquad$ for $t = 1 \ldots T-1$

3. $b_j'(k) = \Sigma\ \gamma_t(j)\ /\ \Sigma\ \gamma_\tau(\varphi)$ $\qquad$ for $t = 1 \ldots T$

The re-estimation formula for π_i is trivially the probability of being in state q_i at $t = 1$. The re-estimation formula for a_{ij} is the ratio of the expected number of transitions from state q_i to q_j, divided by the expected number of transitions out of state q_i. The re-estimation formula for $b_j(k)$ is the ratio of the expected number of times being in state j and observing

Forward and Backward Sweep of the Lattice

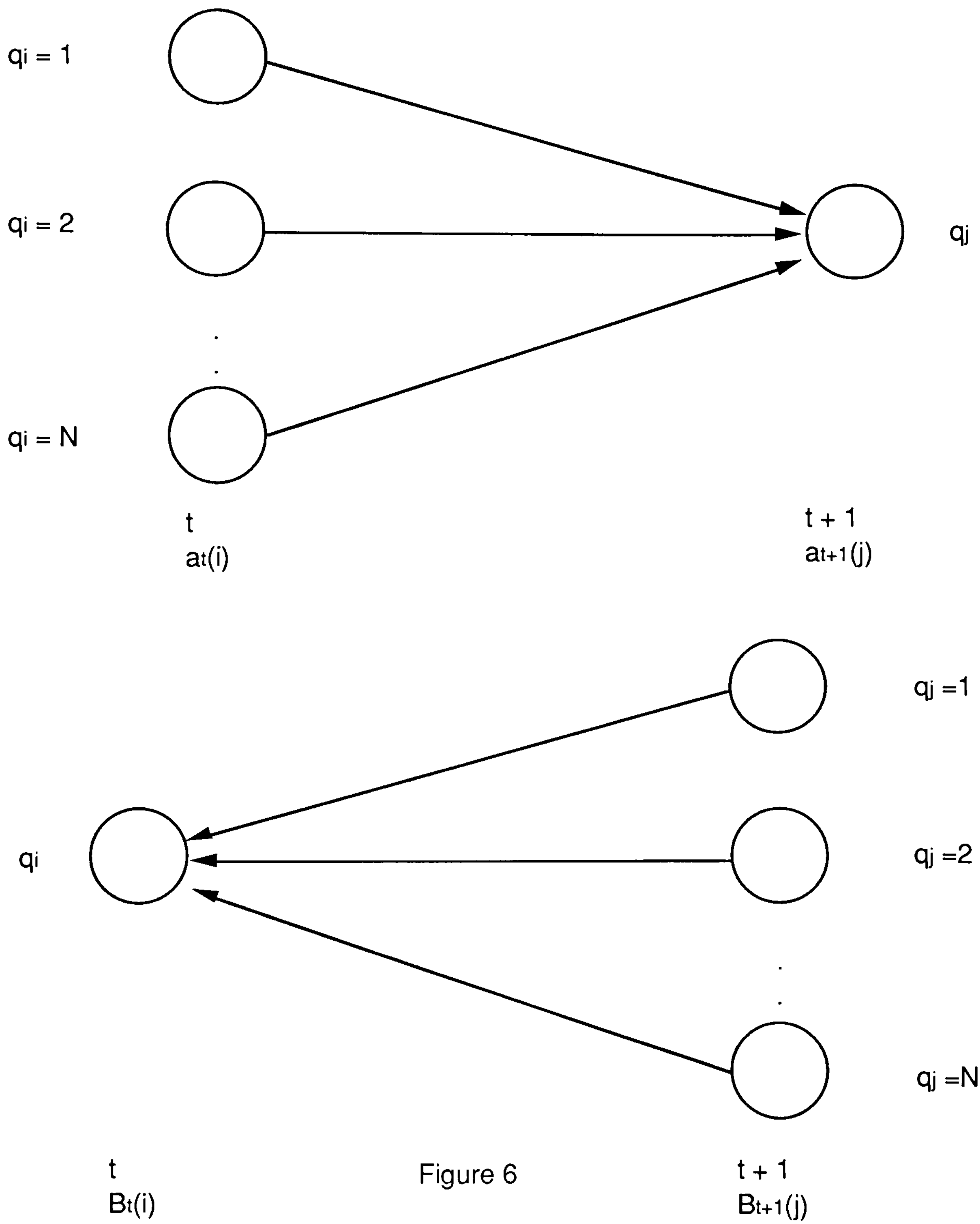


Figure 6

Viterbi Algorithm

Step 1 - Initialization

$\delta_1(i) = \pi_i b_i(O_1)$, $\quad 1 <= i <= N$

$\Psi_1(i) = 0$

Step 2 - Recursion

for $2 <= t <= T$, $\quad 1 <= j <= N$

$\delta_t(j) = \max [\delta_{t-1}(i)\ a_{ij}]\ b_j\ (O_t)$ $\quad$ for $1 <= i <= N$

$\Psi_t(j) = \text{argmax} [\delta_{t-1}\ (i)\ a_{ij}]$ $\quad$ for $1 <= i <= N$

Step 3 - Termination

$P^* = \max [\delta_T(i)]$ $\quad$ for $1 <= i <= N$

$i_T^* = \text{argmax} [\delta_T(i)]$ $\quad$ for $1 <= i <= N$

Step 4 - Path (state sequence) backtracking

for $t = T-1, T-2, \ldots\ldots, 1$

$i_t^* = \Psi_{t+1}(i_{t+1}^*)$

Figure 7

symbol k divided by the expected number of times of being in state j. The primed values of the new model are iteratively used to generate a new model until an optimum point is reached [RABI86].

1.3.4 HMM Types

Hidden Markov Models are categorized by types. As shown in Figure 8, the two primary types of HMM's are ergodic and non-ergodic.

Ergodic HMM's are where all states in the model are interconnected. Using automata terminology, they can be thought of as a clique of states.

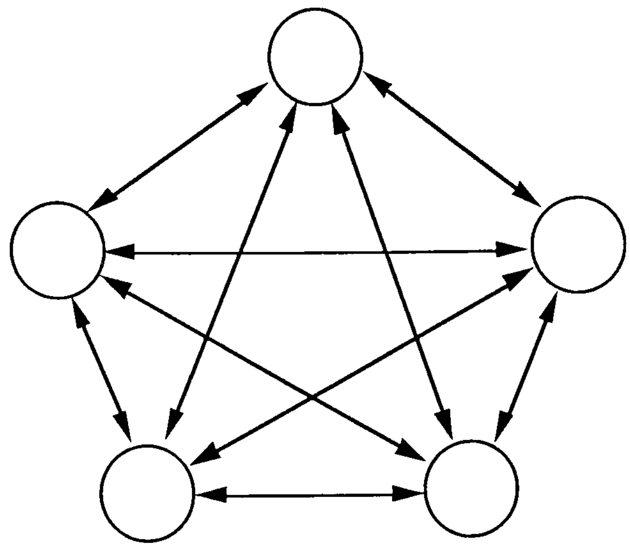
The non-ergodic, and the left-to-right in particular, HMMs are of special interest in speech recognition. The left to right HMM displays an inherent temporal structure. Transitions are allowed only to an equal or higher numbered state. This makes the left-to-right HMM virtually ideal for for modelling time-varying processes such as speech.

2. Project Description and Methodology

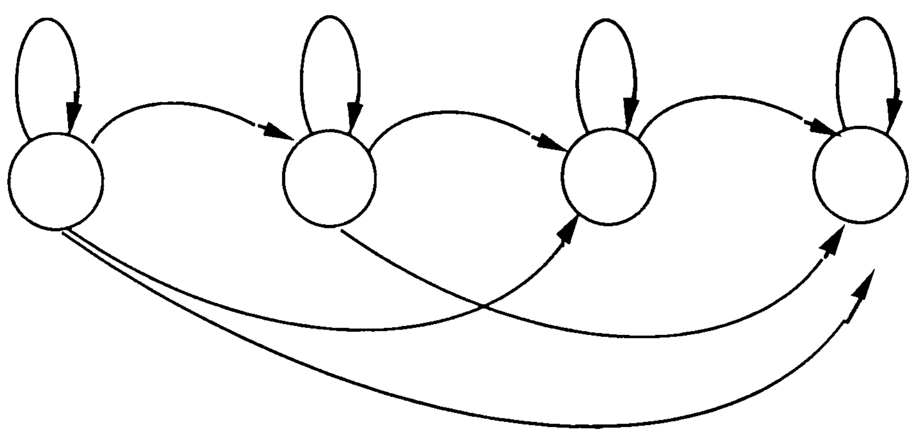
Little or no training data existed for the lexicons available. Therefore, training of the HMM was not a viable option. Hence, the direction of this study was guided toward the decoding problem of HMMs. This study was not used in conjunction with any other part of the speech project. As such, a unique method of generating phoneme strings needed to be developed. In addition, the natural language understanding portion of the speech project required not only the best state sequence, but also required a set of likely alternatives to the recognized phrase. Consequently, a lattice of possible words was constructed.

The implementation of the HMM work was accomplished in three phases. Phase I was the implementation of the baseline HMM system for digits. Phase I included the construction of the phone and word models. The phone models were developed using output probabilities taken from the confusion matrix generated in R. Thomas Selman's thesis [SELM89]. The confusion matrix values originated by taking data from the RIT Research Corporation's front-end vowel classification [HILL87] and from studies of human confusability [SHEP80]. Also included in

Types of HMM's



ergodic



non-ergodic
(left- to-right)

Figure 8

Phase I was the Viterbi search algorithm (to include a beam search capability), variance of error rates in the phoneme models, variance of string length, and the construction of a lattice of different parsings.

Phase II involved the addition of knowledge to the baseline system. A grammar was introduced to the baseline system in an attempt to increase word accuracy.

Phase III involved the scaling up of the baseline model to the cockpit lexicon used in the speech project at the RIT Research Corporation. A grammar, in the forms of word pair and bi-gram which will be explained later, was introduced to the cockpit lexicon to determine its effect on the system.

2.1 Phoneme confusion data

The work performed in this study is based on the assumption that the confusion probabilities used between phonemes were accurate. The vowel confusion probabilities were obtained directly from the vowel classification study [HILL87]. Diphthongs were broken down into combinations of vowels. For example, the diphthong *ay* was represented by *ah ih* . A full listing of the diphthong representations will be presented later. However, for consonants, no confusion data existed. The only data available was the perceptual distance measures between consonants. R.T. Selman's [SELM89] examination of confusion probabilities and distances for vowels yielded an approximation to the following exponential relationship (base ≅ 1.50):

$$\text{Confusion Probability (input vs. output)} \approx \frac{\dfrac{1}{\text{distance}}}{\sum \dfrac{1}{\text{distance}}}$$

where the Σ is taken over all phones.

The above relationship was applied to all phonemes. The confusion probabilities in Appendix A reflect the values obtained. Different values for the confusion probabilities would certainly cause different results in the performance of the HMM.

2.2 Knowledge Used

Inter-word transition probabilities were extracted from the grammars introduced. The word pair grammar, which is a simple grammar that specifies only the list of words that can legally follow any given word, and the bi-gram grammar were used.

The use of word pair knowledge comes from the development of inter-word transition probabilities. These probabilities are a result of the extraction of pairs of words which occur in phrases throughout the lexical domain. The lexical domain is parsed to produce the pairs of words. Then all duplicate word pairs found therein are removed. When a word pair is found, a transition probability is assigned. When no word pair is found, no transition is permitted between the two words. The transition probability used is the inverse of the number of word pairs found for any one word. This implies that the transitions allowed from one word to any other permissible word have the same probabilities. When knowledge is available about the manner in which the lexicon is used, the measure of the number of choices at each decision point can be reduced. The perplexity or entropy of the model is roughly the number of choices per decision point and is a measure of the constraint imposed by the grammar or the level of uncertainty given the grammar [LEE89]. The perplexity of the digit lexicon using no grammar was 10. The perplexity was reduced to 5 when using a word pair grammar. The perplexity of the cockpit lexicon test data set with no knowledge added was 666. The word pair grammar reduced the perplexity to approximately 4.6.

The addition of bi-gram knowledge is similar to that of word pair. The difference is that duplicates are not removed and a more accurate inter-word transition probability is assigned. The probabilities are estimated by counting. The probability is calculated by a division of the number of instances of a specific word pair by the total number of transitions permitted from the first word to any other word. The perplexity of the digit lexicon using a bi-gram grammar was approximately 4.9. The test set perplexity obtained from using bi-gram probabilities on the cockpit lexicon was reduced to approximately 4.4. The limited test data for the cockpit domain did not significantly reduce the perplexity as was the case in the Sphinx speech recognition system. However, better information about inter-word transitions should produce more accurate results.

2.3 Software Tools

A software module was created to facilitate the development of the HMM models. The software module was implemented in COMMON LISP due to the predetermining fact that the RIT Research Corporation Speech Project is being developed on a LISP machine. LISP is a functional programming language oriented for the manipulation of symbols. Using LISP in an interpretive fashion allowed rapid prototyping. This gives the programmer quick conformation of the success or failure of the code. COMMON LISP was developed in an effort to combine the features of other dialects in an optimal was and to promote the commonality among diverging new LISP dialects [STEE84]. This study was developed using the flavor system, which is an object-oriented programming facility. When a flavor type is defined, a data type and a set of operations implemented by function objects called methods that operate on that data type are defined. Instances of these data types can then be generated. The flavor system allows the combination of various flavor definitions to construct a new flavor. This may prove helpful in the integration of several levels of the speech project. The software module contained the following items:

1. Phone and Word model construction
   input - lexicon, error rates, and optional grammar
2. Phoneme String generator
   input - word or phrase sequence, lexicon, and error rates
3. Viterbi search with optional beam search capability
   input - list of word models and phoneme strings
   output - word lattice generated from the search space
4. Grammar construction
   input - collection of sentences or phrases
   output - word pair and bi-gram probabilities
5. Error Analysis
   input - list of correct parsings and search space lattice
   output - performance statistics and the correct % of phrases contained in the search space

Figure 9 shows the general architecture of the HMM system implemented. The model is loaded by using the lexicon, error rates and an optional grammar. The phoneme string is then generated and used as an input to the Viterbi module. Hypothesized words are produced and statistics are gathered.

2.3.1 Phone and Word model construction

The model used to represent words was one of parallel connected phoneme strings. The baseline digit model is shown in Figure 10 . This model was used for its simplicity. Once a starting state was determined, transitions between states occurred in a very clean and easy to follow manner. With no grammar imposed, a transition from a terminal state to an initial state occurs with a probability equal to the reciprocal of the number of words in the lexicon.

Deletion errors were permitted as indicated in Figure 10. Insertion errors occurred at their specified error rate and simply resulted in a return of the model to the same state it was in last. Insertions are not shown in Figure 10. Deletions from one state to a state located 2 temporal positions to the right in the model took place with a probability equal to the specified error rate. Any deletion transition greater than 2 temporal positions had an associated probability which was decreased as the square of the number of positions jumped above 2 . While tolerating transitions of great magnitude, this particular modeling structure imposed low probabilities to preclude this from frequently occurring.

With an optional grammar introduced, the transitions permitted from any terminal state to any other initial state were constrained. This effect reduced the expected number of inter-word transition decisions in the model significantly.

2.3.2 Phoneme string generator

In all three phases of this thesis, phoneme strings were generated by the software module using the confusion matrix data (Appendix A) and a random number generator. Cumulative output and transition probabilities were maintained to facilitate the production of a

General Architecture

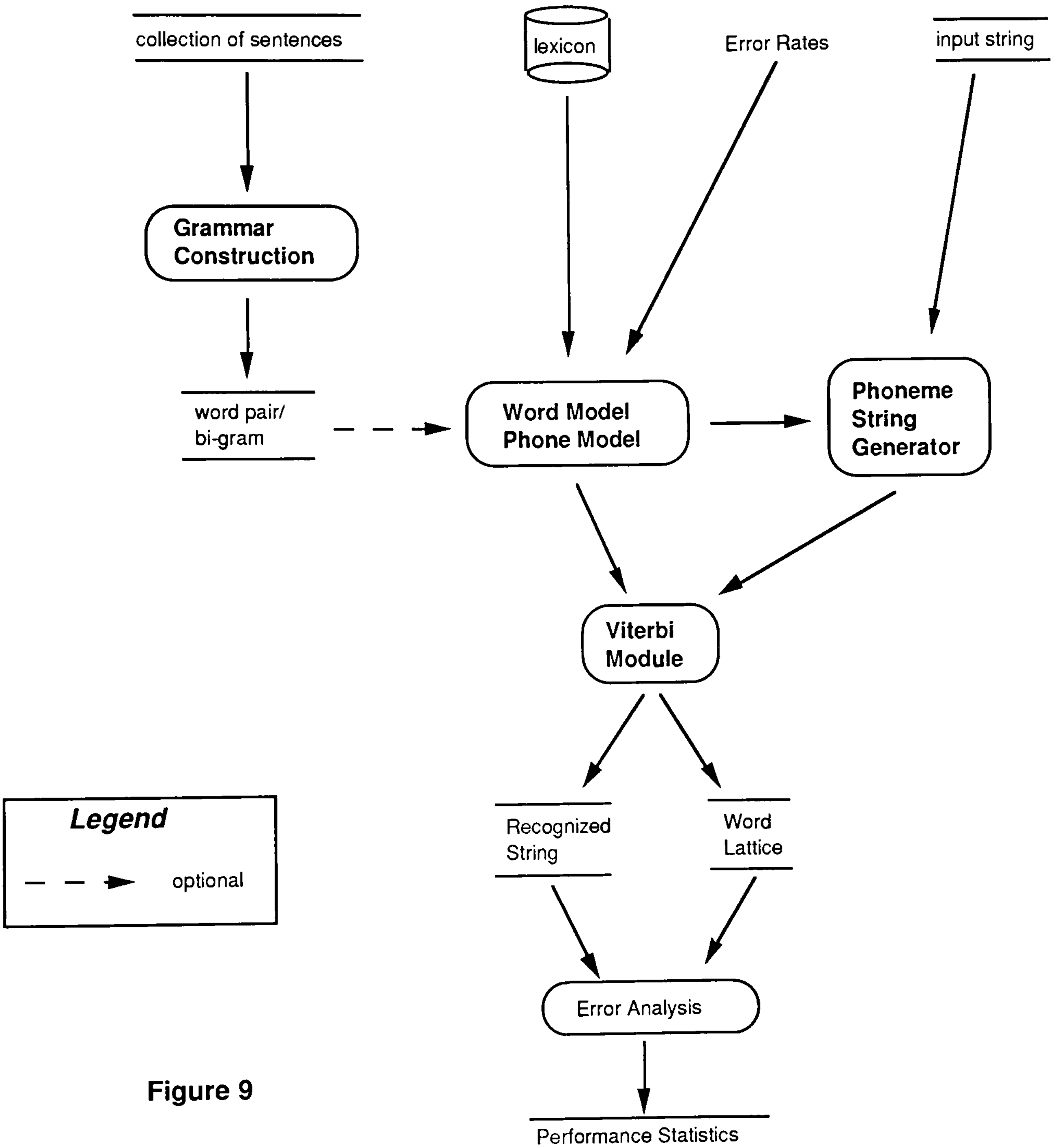
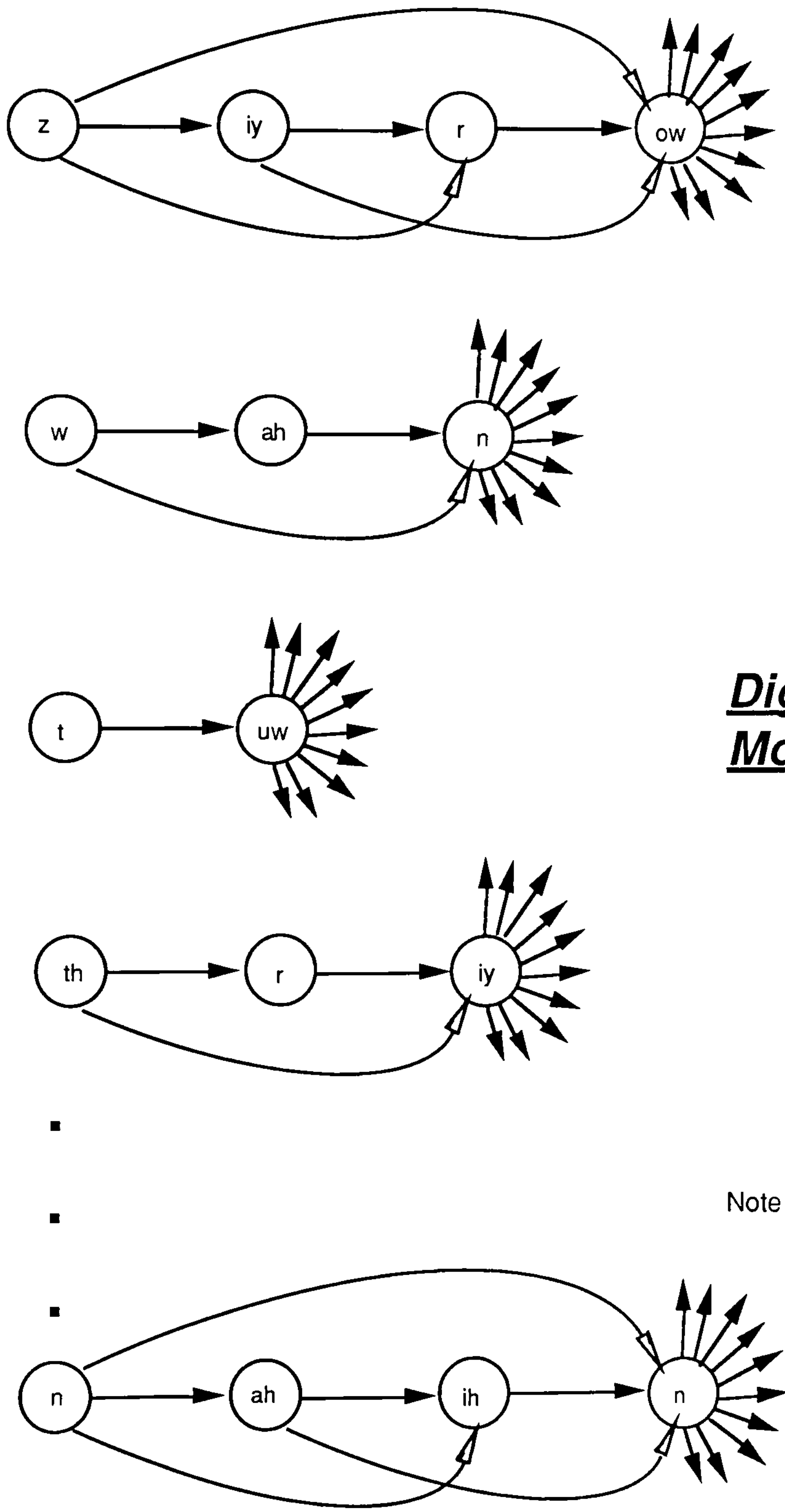


Figure 9



Digit Model

Note: insertion transitions are not shown

Figure 10

phoneme and the movement within the model to the next state in time. A random number was generated. When the magnitude of the random number was less than or equal to the cumulative probability, that phoneme was instantiated and appended to the phoneme string. The same procedure was repeated to determine the transition to the next state.

During Phase I, the starting location as well as the transition between word models was determined at random. This produced truly random sequences of words. The only restriction placed upon the construction of a phoneme sequence was that it terminate after a preset number of phoneme based word strings were created. Word phrases of length between 1 and 5 were used to create phoneme strings.

The addition of a grammar to the baseline digit system in Phase II affected the production of a phoneme string. The transitions between a terminal state of one word and an initial state of another word were now no longer randomly generated, but were constrained by the inter-word transition probabilities obtained from the grammar.

In Phase III, the generation of phoneme strings was further restricted. The phoneme generator determined the starting location based on the first word of the input word sequence. Transitions were permitted as before until a transition to another word was reached. Then the next word in the input sequence was taken and the procedure was repeated. The testing of specific word phrases from the cockpit lexicon did not permit the generation of a phoneme string randomly or as in Phase II.

2.3.3 Viterbi search with optional beam search

The beam search option of the Viterbi algorithm restricts the allowable space to search for the correct phrase [LEE90]. With no beam search capability, the Viterbi algorithm looks at every possible state at any particular instant in time to determine the most likely state (Figure 11). For large vocabularies, the exhaustive search technique without the use of a grammar can be a time consuming and resource expensive process. The beam search option restricts the search space that the Viterbi algorithm is permitted to look at by pruning off unlikely candidates. Two versions of the beam search were used on the digit model. Using a simplistic approach, a strict linearly bounded search space was used by evaluating only the top number of

Lexical Search Space

with no beam search

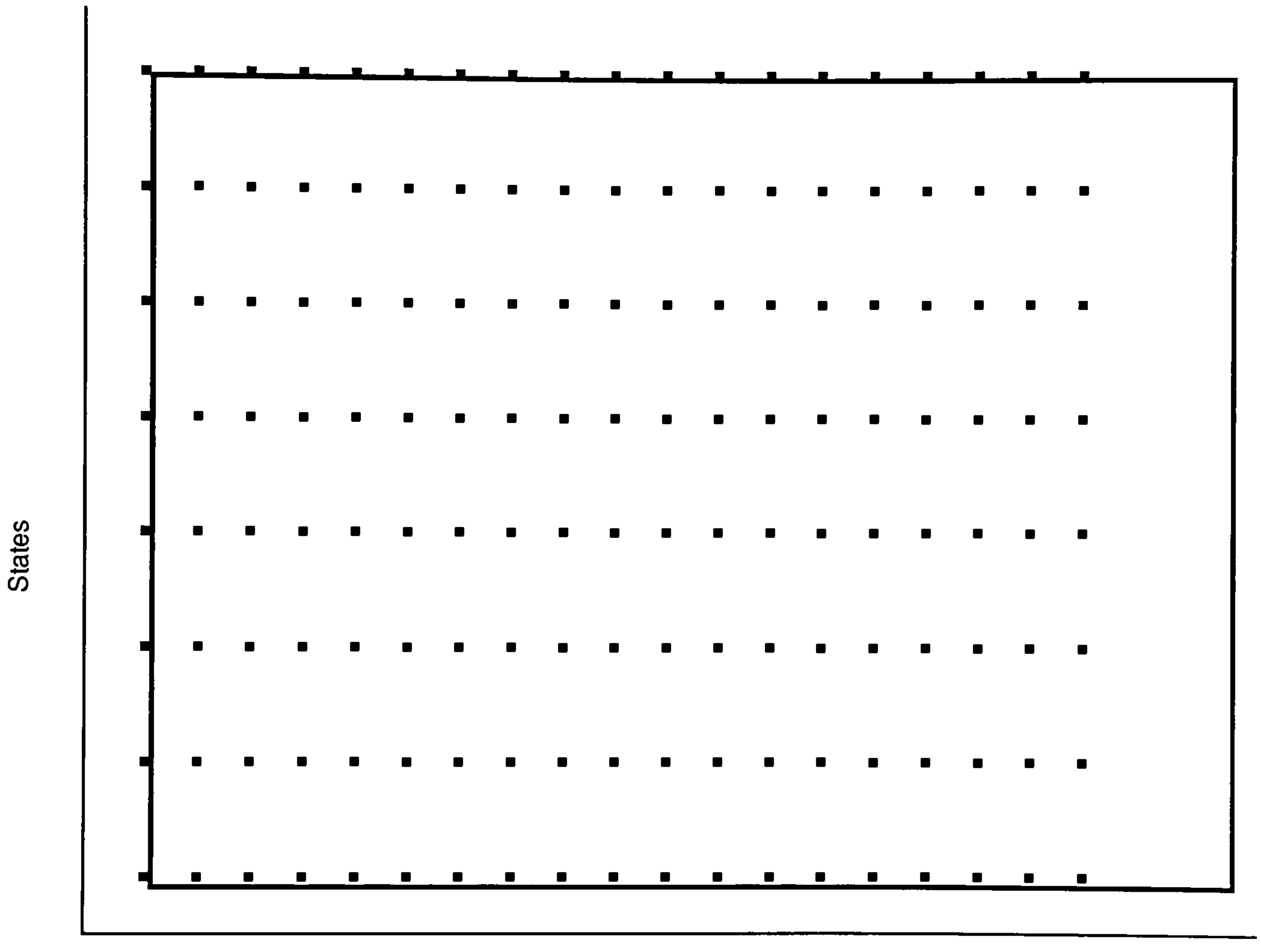


Figure 11

δ values and their associated state numbers (Figure 12). This figure is a crude representation of the search space which is very difficult to portray. In fact, there may exist more than one set of boundaries inter-dispersed throughout the lattice. The second version was to expand the search space initially and then to decrease the search space linearly after several transitions through the lattice structure (Figure 13). This technique was used to prevent the correct phrase from being pruned at an early stage in the lexical search space.

As part of the Viterbi module, a lattice of possible words found within the search space was constructed. This lattice generated will be passed on to the next higher level of the speech project for natural language understanding. Along with the word identified, the lattice contained the starting and ending position in the phoneme string and the probability of the word.

2.3.4 Grammar construction

The grammar used as a knowledge source in the digit model allowed transitions only between words that were even and between words that were odd. As a result, only word strings like *zero-six-two* and *one-nine-three-five* could be generated. Additionally, a transition back to the same word was twice as likely as a transition to a different word. This would make generating the phrase *zero-zero-zero* more likely than that of *zero-two-zero*.

The grammar used in the cockpit model is an amalgamation of data taken from three different sources. Situational input phrases taken from the Cockpit Natural Language Study [LIZZ87], phrases used for evaluation purposes by R.T. Selman [SELM89], and additionally constructed phrases were used. The phrases used for evaluation purposes served as a basis for comparing the results of R.T. Selman's dynamic time warp work and the results of this study. The additionally constructed phrases were incorporated to ensure that every word in the lexicon had a transition probability associated with its terminal states.

In both sets of grammars, sentences or phrases were used as input. As mentioned earlier, in determining word pair probabilities, all duplicate word pairs were removed from the input. They were left in for the calculation of bi-gram probabilities. The entire grammars used for the digit and cockpit lexicons are identified in Appendices B-1 and B-2, respectively.

Lexical Search Space

with linear bounds

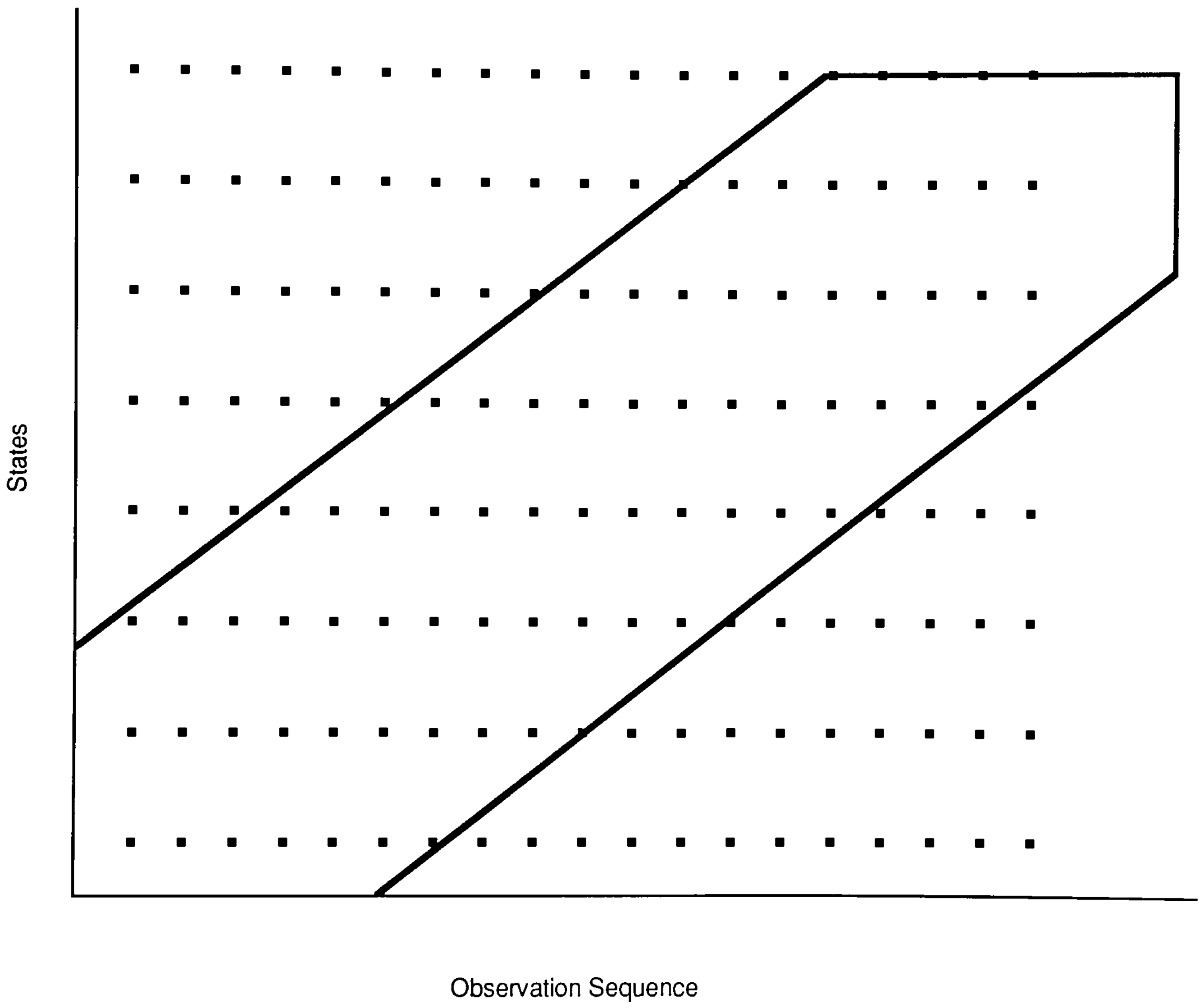


Figure 12

Lexical Search Space with extended and then linearly reduced boundaries

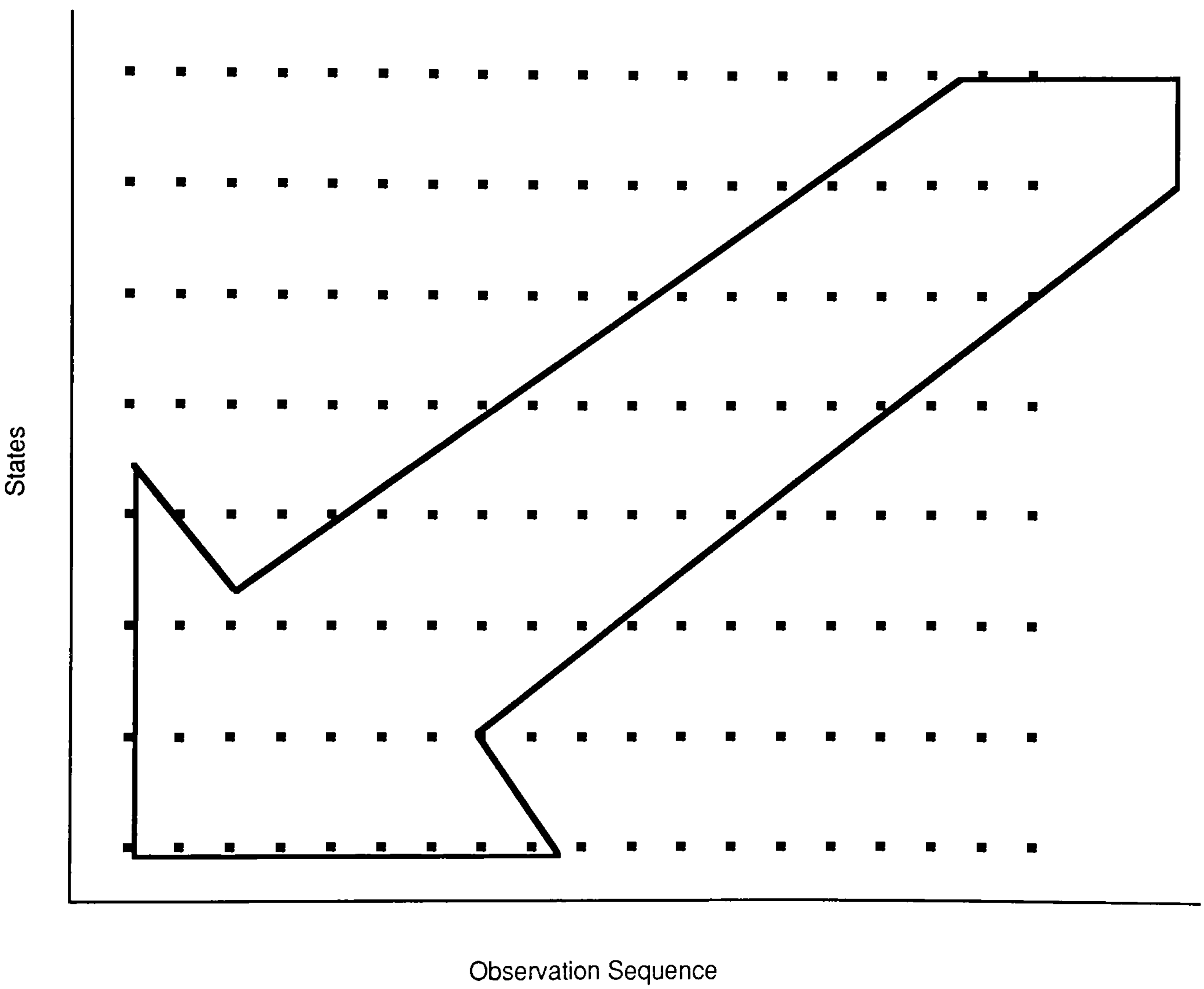


Figure 13

2.3.5 Error Analysis

In all three phrases, the percentage of correct phrases was determined. Also, the lattice structure of possible words contained in the search space generated as a result of the Viterbi module was used to determine if the correct phrase was present in the search space. The percentage of occurrences of the correct phrase in the search space was also calculated. This will be used as an indicator of the results that the natural language portion of the speech project can expect. The average time required to process phoneme strings through the Viterbi module was also recorded for each phase of this study.

2.4 Hardware tools

This study was implemented on a Texas Instruments Explorer[3] II machine. The Explorer is a microprogrammed, dedicated workstation, providing a comprehensive Artificial Intelligence (AI) environment for fast symbolic processing. Additionally, the Explorer can be augmented with a TMS 32020 Signal Processor Board that allows low-level feature extraction to proceed in parallel using four independent signal processors. These characteristics allow the integration of low-level processing with the high level control mechanisms typically found in AI applications.

2.5 Lexicon construction

The vocabulary used in this study was taken from the United States Air Force Cockpit Natural Language study [LIZZ87]. For each of the 656 words contained in the study, a lexicon entry was constructed containing the word in the study and its phonetic transcription. The words from the Air Force study were input into a text-to-speech synthesis system as part of R.T. Selman's study. The output from this system was converted to reflect the Carnegie-Mellon

University phonetic symbol-set (Figure 14). Homonyms form a single lexical entry with multiple English representations. Words with multiple pronunciations were only entered once into the lexicon. For those words, a proper phonetic transcription was taken from Webster's New World Dictionary [WEBS66] and then entered into the lexicon. As mentioned earlier, diphthongs were broken down into their vowel components. The digit and cockpit lexicons are located in Appendices D-1 and D-2, respectively.

2.6 Test data creation

The Air Force Cockpit Natural Language study [LIZZ89] was the source of the test utterances used as input strings for the HMM process. A set of 148 test phrases was selected from the study that combined a wide variety of words available from the lexicon. The average length of the phonetic transcription over the 148 phrases was 20.4. Initially, test phrases were translated to their phonetic representations with a 5% substitution error and 0% insertion and deletion error rates. After this, insertion and deletion errors were increased by intervals of 5% until their maximum of 20% was reached. The substitution error rate was then increased in intervals of 5% and the process was repeated. The errors in the phonetic transcription were created as mentioned earlier. It was believed these error rates would be more than sufficient to represent actual errors obtained from the phonetic classifier.

2.7 Log probabilities

In a lexicon of substantial size, the probability of a word, phrase or sentence rapidly approaches zero during forward computations in the Viterbi algorithm. Recall that the calculation of the δ value results from the multiplication of the previous δ value and the transition probability a_{ij}. Normally, this would result in a floating point underflow. To preclude an underflow condition, the probabilities were transformed into log probabilities. Now instead of multiplying two numbers together, we simply add their logs.

CMU Phoneme Symbols

Vowels	Example
er	bird
aa	cot
ae	bat
ah	butt
ao	bought
aw	bough
ax	the
eh	bet
ih	bit
ix	roses
iy	beat
ow	boat
uh	book
uw	boot

Liquids	Example
l	led
r	red

Glides	Example
y	yet
w	wet

Stops	Example
ng	sing
t	tot
d	dad
k	kick
g	gag
m	mom
n	non
p	pop
b	bob

Fricatives	Example
hh	hay
f	fife
v	verb
th	thief
zh	measure
dh	they
s	sister
z	zoo
sh	shoe

Affricates	Example
ch	church
jh	judge

Diphthong Conversions		
ay	->	ah ih
ey	->	eh ih
oy	->	ow ih

Figure 14

The addition of logs is required when normalizing a matrix of log values. The addition of logs is more complicated. For us to add two numbers P_1 and P_2 where P_1 is greater than or equal to P_2, the following formula taken from the Sphinx system was used.

$$\log_b (P_1 + P_2) = \log_b [\, b^{\log b\ P1} + b^{\log b\ P2}]$$
$$= \log_b [\, b^{\log b\ P1} (1 + b^{\log b\ P2 - \log b\ P1})]$$
$$= \log_b P_1 + \log_b [1 + b^{\log b\ P2 - \log b\ P1}]$$

3 Results

The accuracy results for all three phases are contained in their entirety in Appendix C and represent the results achieved from testing both lexicons over the previously mentioned range of error rates.

Below are three examples of the results produced by the software modules. The first two are taken from Phase I. The third example is taken from Phase III.

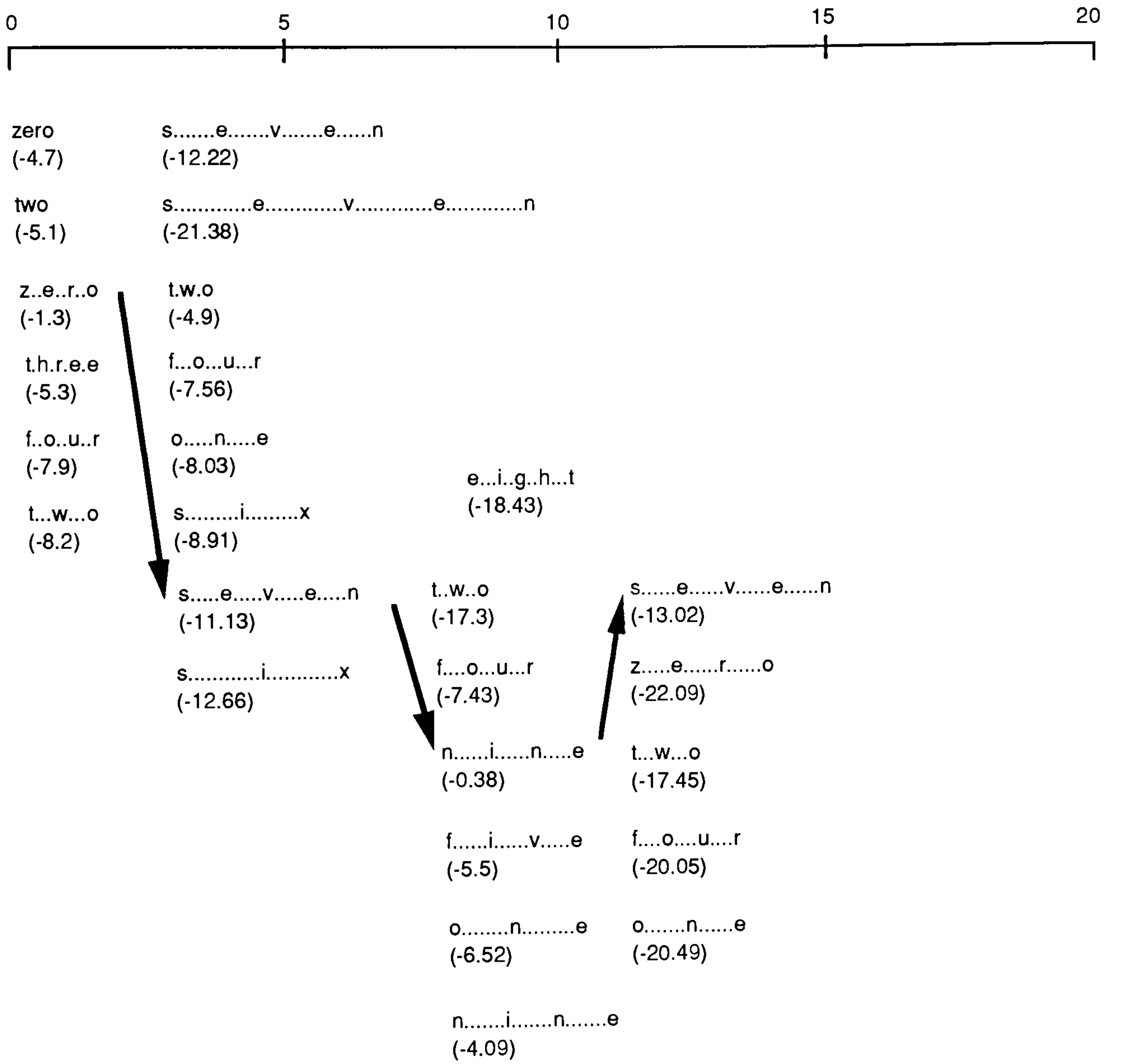
The first example illustrates a situation where the correct phrase was not found as the most likely candidate. However, the correct phrase is located in the word lattice. The second example shows where the correct phrase was not found in either the Viterbi search or the word lattice. The third set of examples shows results obtained from the Viterbi module in Phase III.

Example Set 1

Actual (zero seven nine seven)
Phoneme string (z r ow s eh d sh n n ah ih n s eh v ax n)
Found (zero six nine seven)

substitution .1
insertion .1
deletion .1

Search Space (phoneme length)

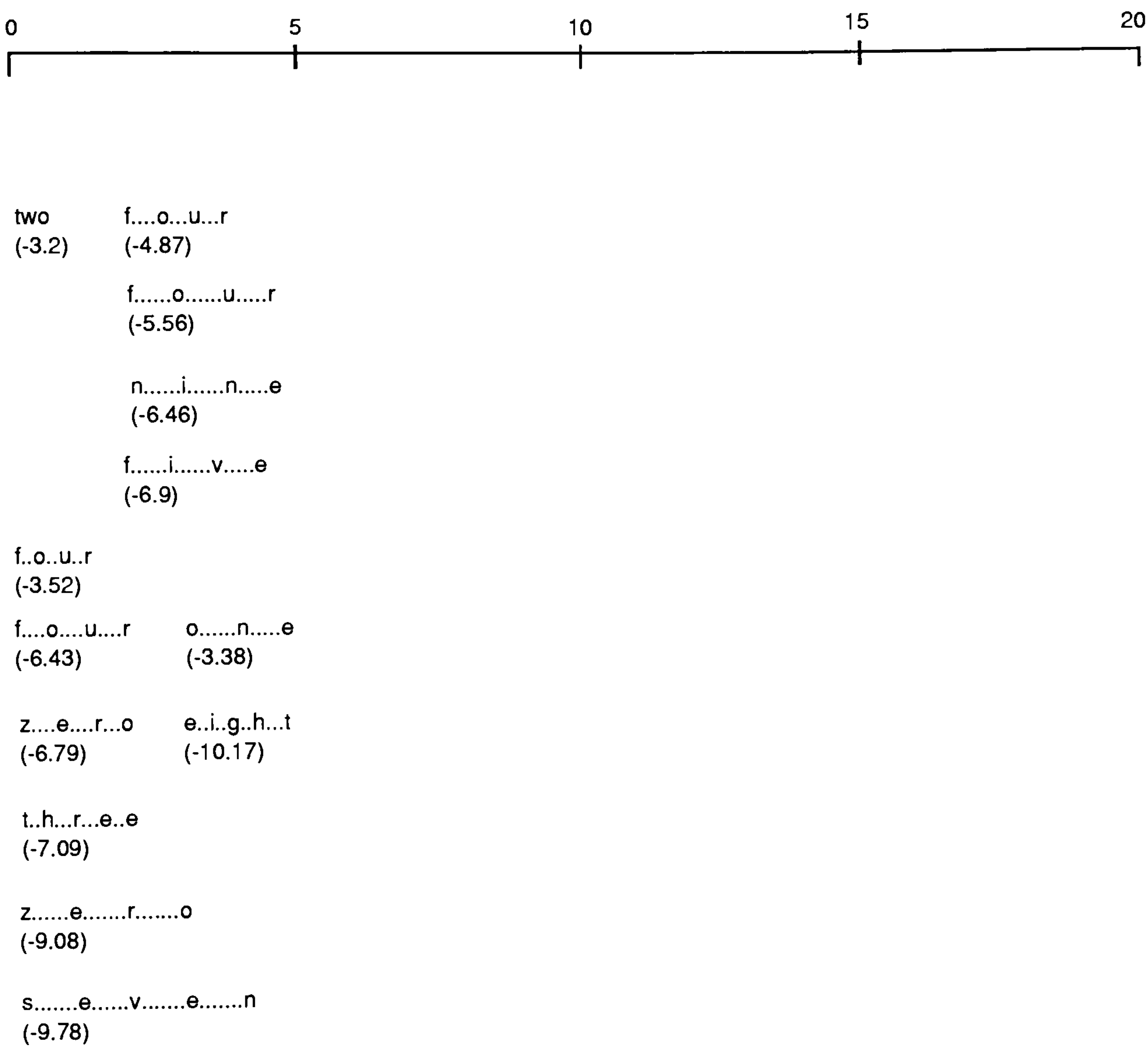


Example Set 2

Actual (five one)
Phoneme string (f l b uh ah r)
Found (four one)

substitution .5
insertion .1
deletion .1

Search Space (phoneme length)



Example Set 3

correct phrase - range and bearing wingman

phoneme string - r ae ih n jh ae l n g b eh ih r ix jh w ih ih ng ng g m uh b

phrase found - range and bearing wingman

correct phrase - status of strike flight

phoneme string - s t eh l ax s ax v s hh aa ih ih k sh l ih ih t

phrase found - status of strike flight

correct phrase - give me more information on the threat

phoneme string - g ih eh v m iy m ow r ix n f er m sh ix f aa aa y g ax th r eh n

phrase found - give me more information on visual

3.1 Phase I

Initially, all candidates with a δ value below the top 5 were pruned off from the search space. Then the search space was expanded to prune off only those with a δ value below the top ten states. Figure 15 shows that for a substitution error of .15, an insertion error of .10, and a deletion error of .10, phrase recognition accuracy was above 70%.

As was expected, the increase in the search space raised the accuracy of the system. However, early pruning caused the correct phrase to be located outside of the search space on several occasions. For that reason, the search space was initially expanded and then reduced after several steps through the HMM lattice. Figure 16 shows the improvement in accuracy caused by the expanded search space.

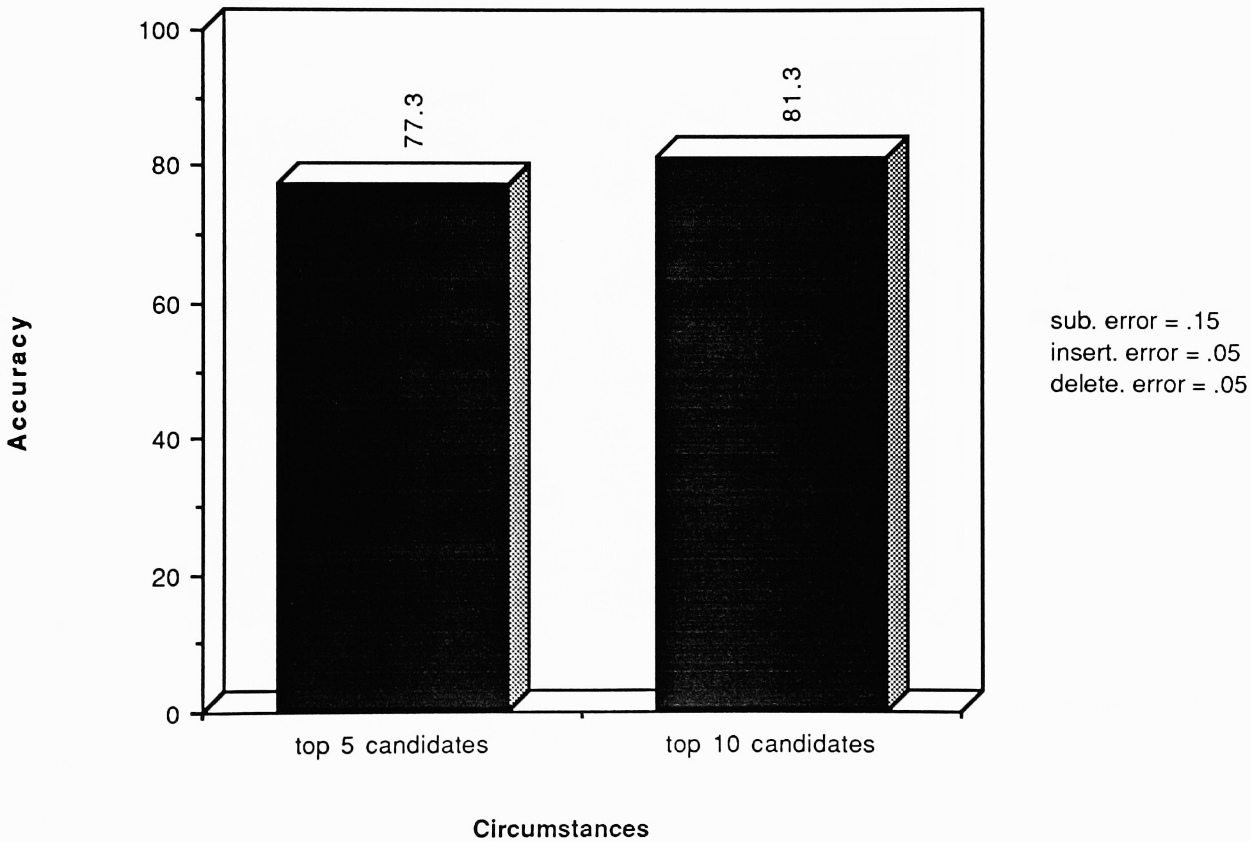
The test results reflect statistics gathered from the testing of phrases of different length (between 1 and 5) words. Each iteration of the particular phrase length specified was run through the Viterbi module 30 times.

The bar graphs shown are indicative of the set of results obtained for the digit lexicon with no grammar. Figure 17 compares other results over the four different circumstances tested. There was a gradual decrease in word phrase accuracy as the error rates increased.

The system ran a phoneme string through the Viterbi module and produced a hypothesized word string in real time. For a lexicon of size 10 and perplexity 10, this was not surprising.

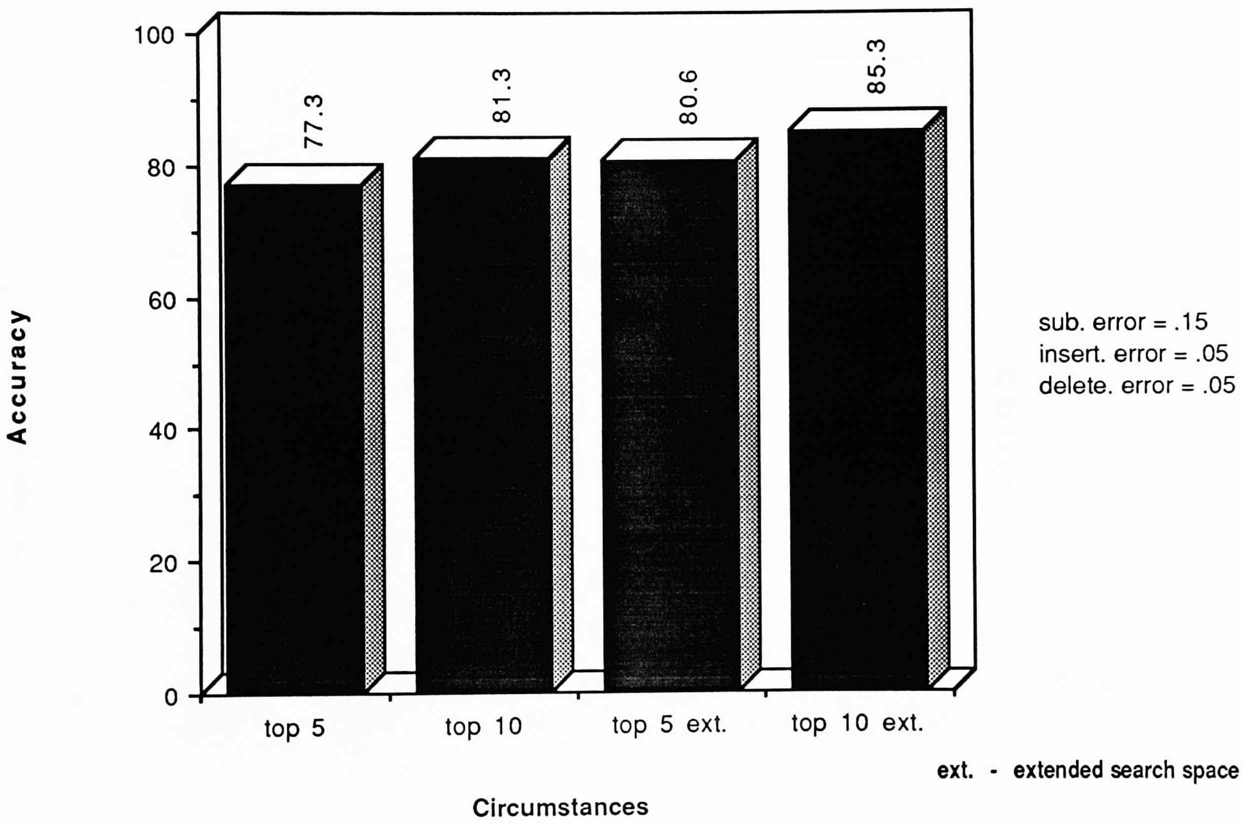
Comparison of Phase I Search Space Parameters

Figure 15



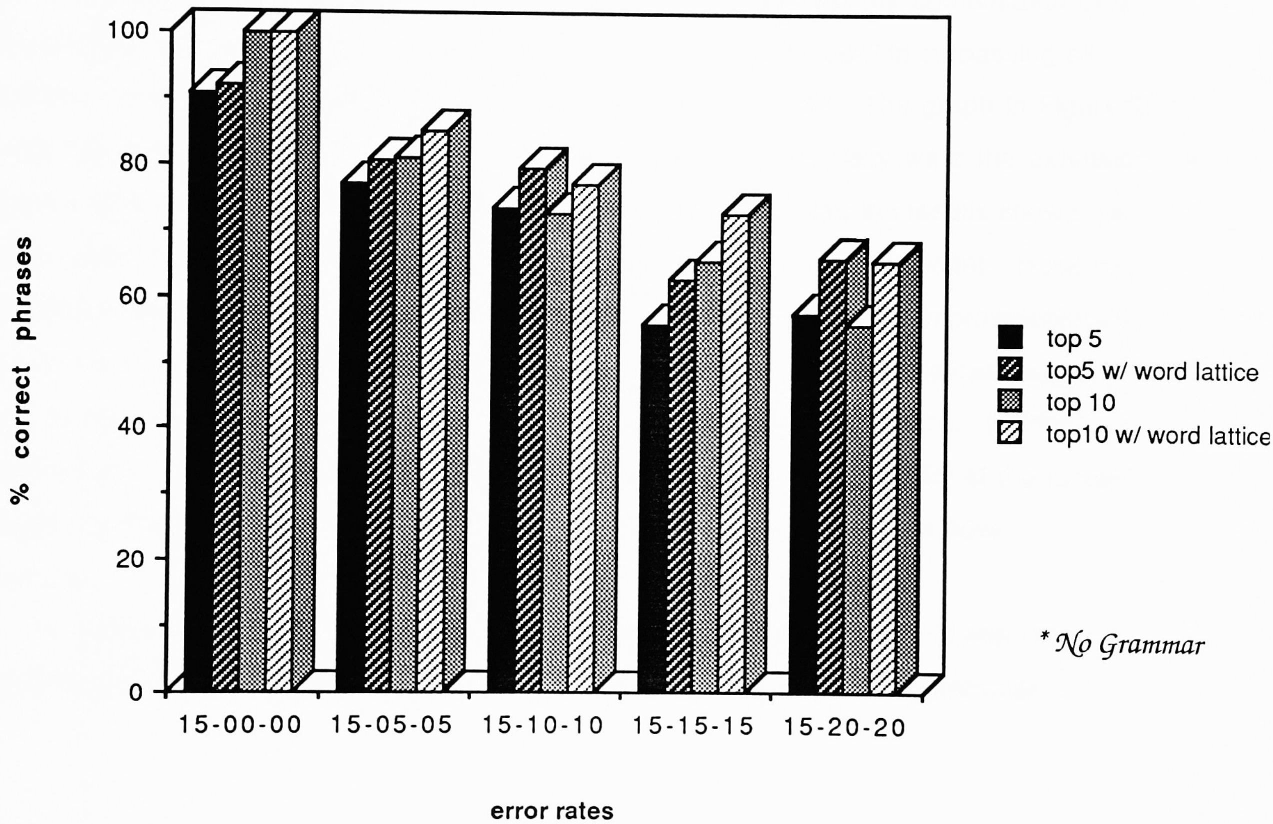
Comparison of Phase I Parameters with an Extended Search Space

Figure 16



Comparison of Various Error Rates
in Phase I

Figure 17

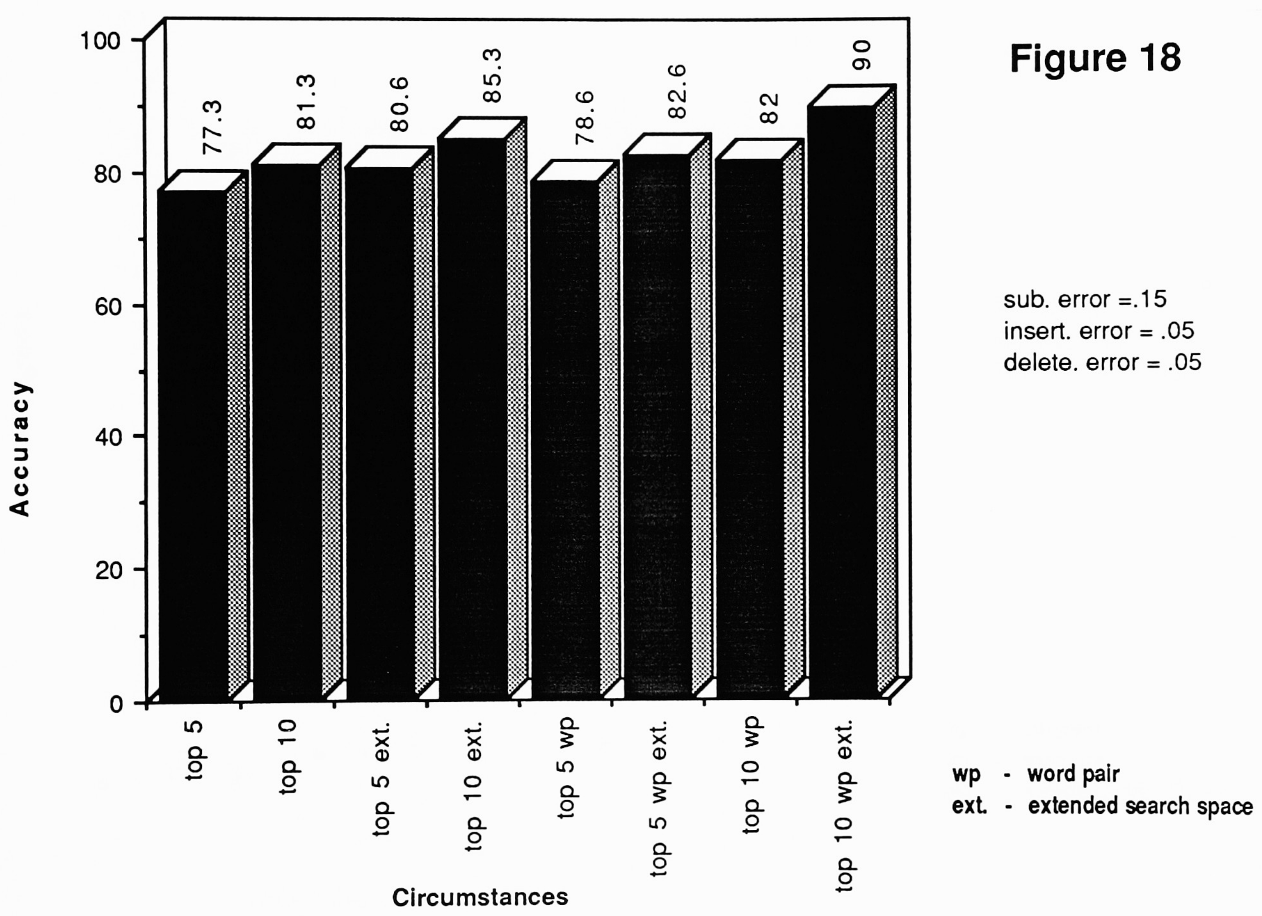


3.2 Phase II

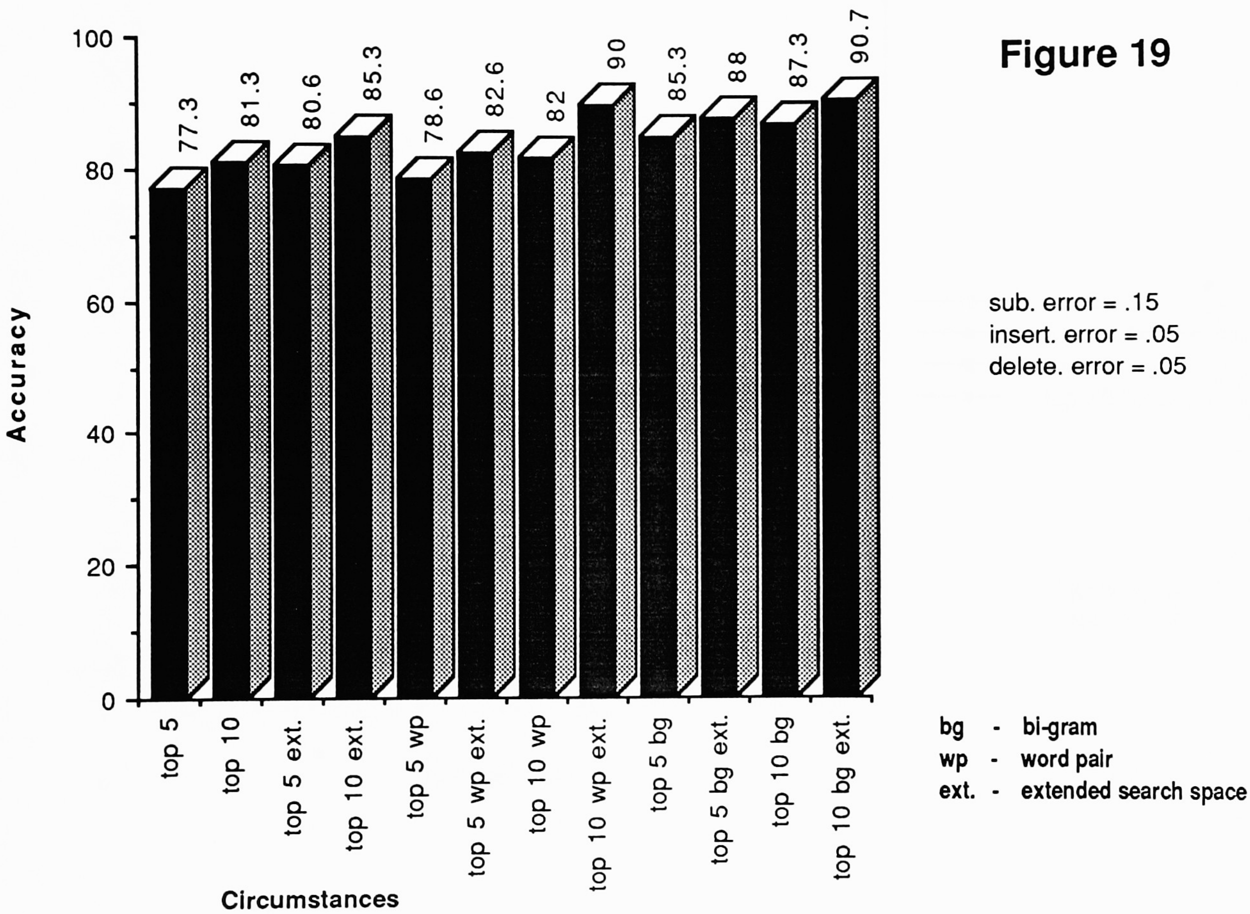
Grammar was introduced to the model in terms of the use of word pair and bi-gram probabilities. The addition of a word pair grammar increased the phrase accuracy above that achieved without knowledge (Figure 18). The combination of an extended search space and word pair grammar provided the best results to that point. Overall, the word pair and bi-gram knowledge improved the performance of the system. However, the increase in performance was not as great as that obtained by Kai-Fu Lee [LEE89]. Lee increased the accuracy of his system nearly two fold by decreasing the perplexity from 997 with no grammar to 60 with word pair grammar and finally to 20 with bi-gram grammar. As Figure 19 shows, the combination of a bi-gram grammar and an extended lexical search space assisted the model in surpassing all other results. These tendencies were consistent through all the results. The graph in Figure 20 shows that the primary variables involved in increasing the model accuracy were the extension of the search space and the addition of knowledge to the system. Because the results shown are taken over all of the error rates tested, the accuracy figures may not be significant. However, an appreciation of the impact of certain variables can be obtained. Dramatic improvements in the accuracy of the system were not realized when grammars were added. This can be primarily attributed to the low perplexity of the lexicon. In Phase I, the perplexity was 10. In Phase II, the perplexity was reduced to 5. Because the drop in magnitude of the perplexity of the model from Phase I to Phase II was not substantial, the accuracy of the system did not raise significantly.

As was the case in Phase I, Phase II saw the Viterbi module operating in real time. Again, this was due to the small perplexity of the model after the addition of a grammar.

Comparison of no Knowledge and the use of a Word Pair Grammar in Phase II

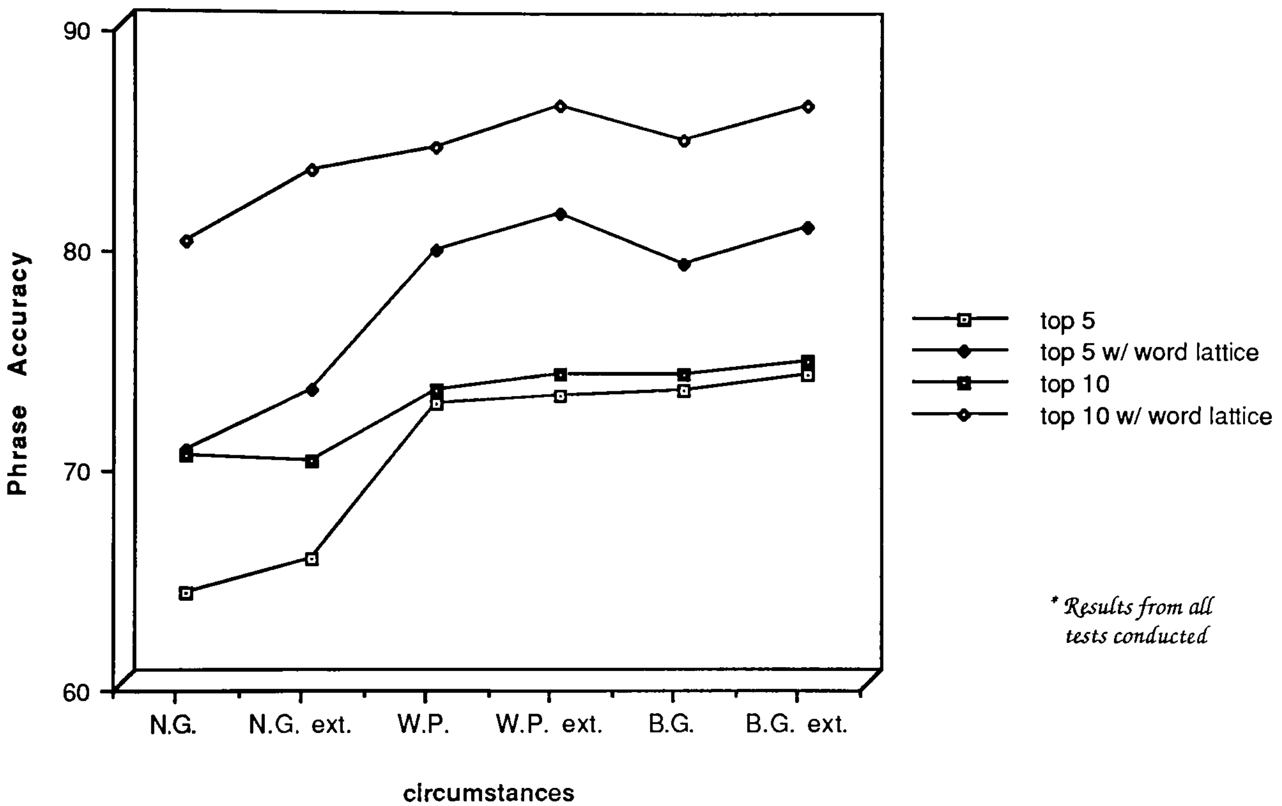


Comparison of no Knowledge, Word Pair and a Bi-gram Grammar in Phase II



Results from Phase I and Phase II

Figure 20



3.3 Phase III

The tests were conducted on the cockpit lexicon using the following search space parameters:

Position in the Lattice	Candidates kept
1	top 100
2 - 5	top 70
> 5	top 35

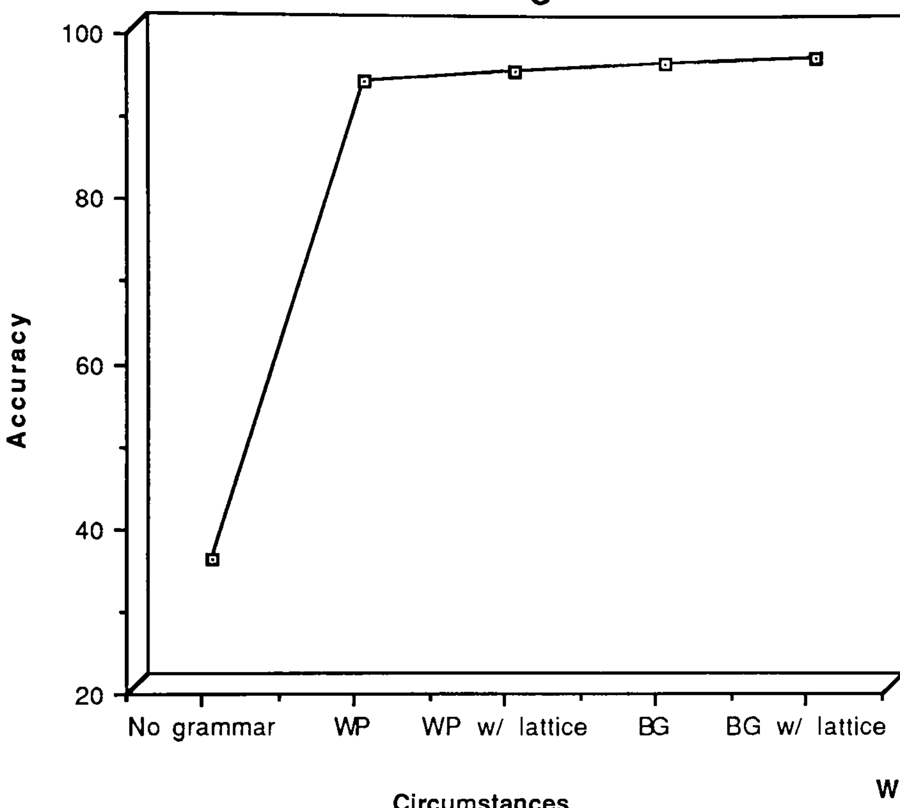
The parameters were set to these values in an attempt to prevent early pruning of the correct candidate.

The cockpit lexicon was tested with no grammar added. Preliminary test results indicated an accuracy of approximately 35% (7 out of 20) for error rates of .05, 0.0, 0.0 for substitution, insertion and deletion. Only preliminary tests were conducted because of the time elapsed when processing a phoneme string. A perplexity of 666 necessitates that at each step through the HMM lattice structure, 666 other possible transitions be considered. The pruning process removed all but the most likely candidates. However, pruning could not occur until after all δ values were calculated and sorted. The end result of having such a high perplexity was that the time required to process only one phoneme averaged about 12 minutes. That corresponds to approximately 4 hours per input string and an overall time of 30,000 hours to test the data over all error rates considered. This time simply proved too great for further practical consideration. Hence, testing without knowledge was discontinued. However, the time required to process an input string using no grammar is not as dismal as it may appear. Levinson conducted tests on a 47 states ergodic semi-Markov model using an 8 CE Alliant FX-80 super computer and a 992 word lexicon [LEVI90]. He found that recognition required about 15 times real time.

With a grammar added, the percentage of phrases more than doubled (Figure 21). A comparison using the error rates of .05, 0, 0 for substitution, insertion and deletion errors respectively, reveals that there was little significant improvement in using a bi-gram grammar verses a word pair grammar. The improvement noted is primarily due to more

Results from Phase III

Figure 21



sub. error = .05
insert. error = 0
delete. error = 0

WP - word pair
BG - bi-gram
w/ lattice - word found in lattice

precise inter-word transition probabilities being assigned. There was a small decrease in the perplexity of the model (~ 4.4). This was caused by the static properties of the model and the small size of the grammar. Probabilities were not adjusted because training of the model was not conducted. The lack of training resulted in no reduction in the number of different transition locations that must be considered.

For substitution error rates at 10% or below, an increase in insertion and deletion error rates caused a monotonic decrease in the system performance. When substitution error rates rose above 10%, an increase in insertion and deletion error rates from 0% to 5% resulted in a sharp drop in the percentage of phrases correctly recognized (Figure 22). The reason for this acute decrease may be caused by the size of the lexical search space. An increase in the number of candidates retained for the next step through the lattice would certainly increase the phrase recognition figures.

An examination of the overall performance (Figure 23) indicates there is a small advantage in using a bi-gram grammar over a word pair grammar. The grammar used (Appendix B-2), although seemingly lengthy, provided only moderate gains in accuracy. This was a result of the inter-word transition probabilities being more precise and a slight decrease in the perplexity.

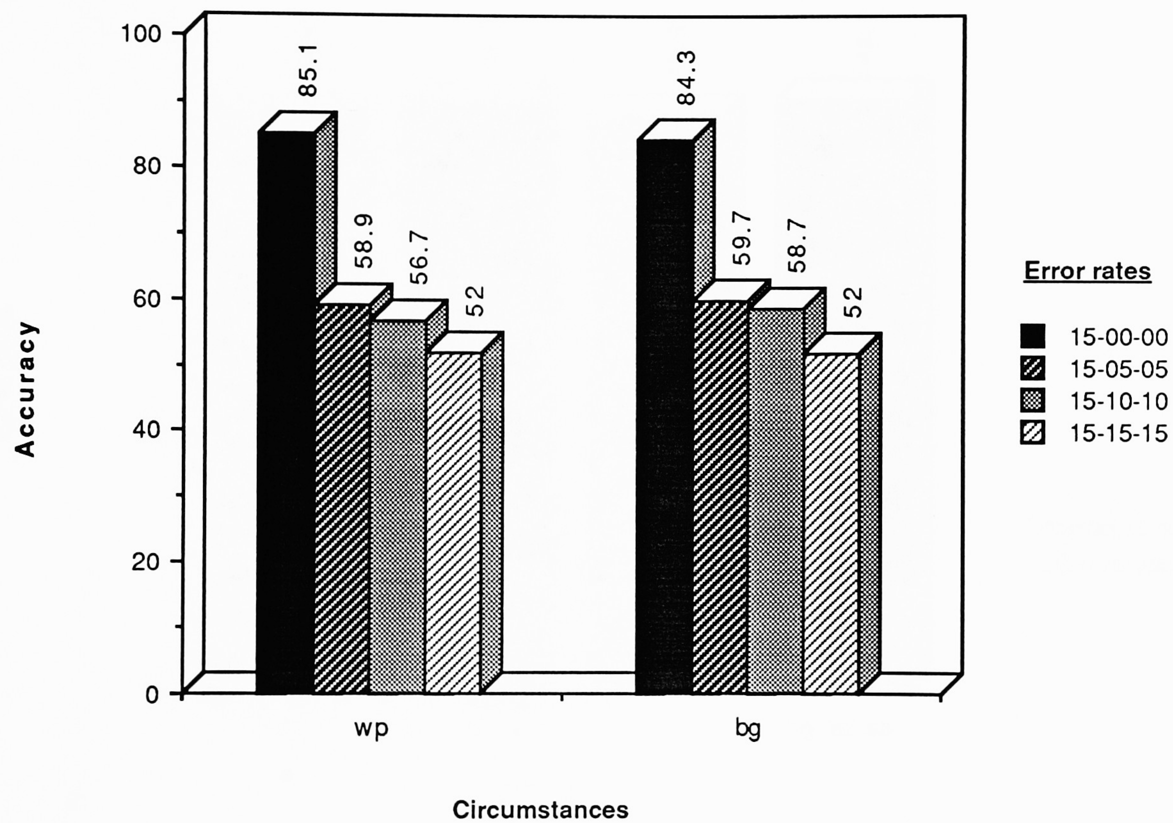
3.4 Comparison of HMMs and Dynamic Time Warping

The results of Hidden Markov Models compare favorably with those of the Dynamic Time Warping methodology used by Selman. As Figure 24 illustrates, when using equal substitution rates, the HMM was able to recognize a higher percentage of the input phrases. While this may not be a fair comparison, it does show that the HMM method permits the incorporation of knowledge while the DTW does not. The incorporation of knowledge about the lexical domain provided enhanced recognition performance. In addition, this illustrates an important feature of HMMs: that knowledge can be easily added to the model.

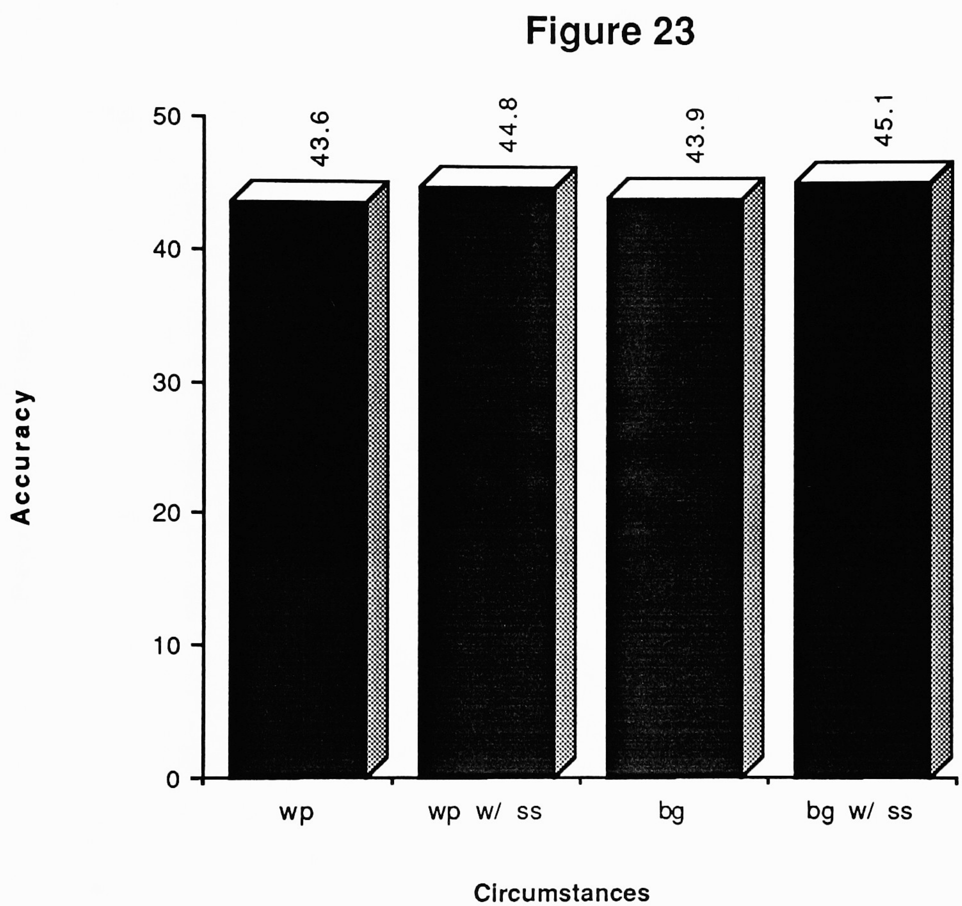
The time required to process a phoneme string for the Dynamic Time Warping method was on the order of 9 seconds per phoneme in test conditions providing the greatest percentage of complete phrase recognition. Approximately 2.5 seconds per phoneme was used by the Viterbi module to recognize the input phrase when using knowledge.

Comparison of Word Pair and Bi-gram over Different Error Rates

Figure 22

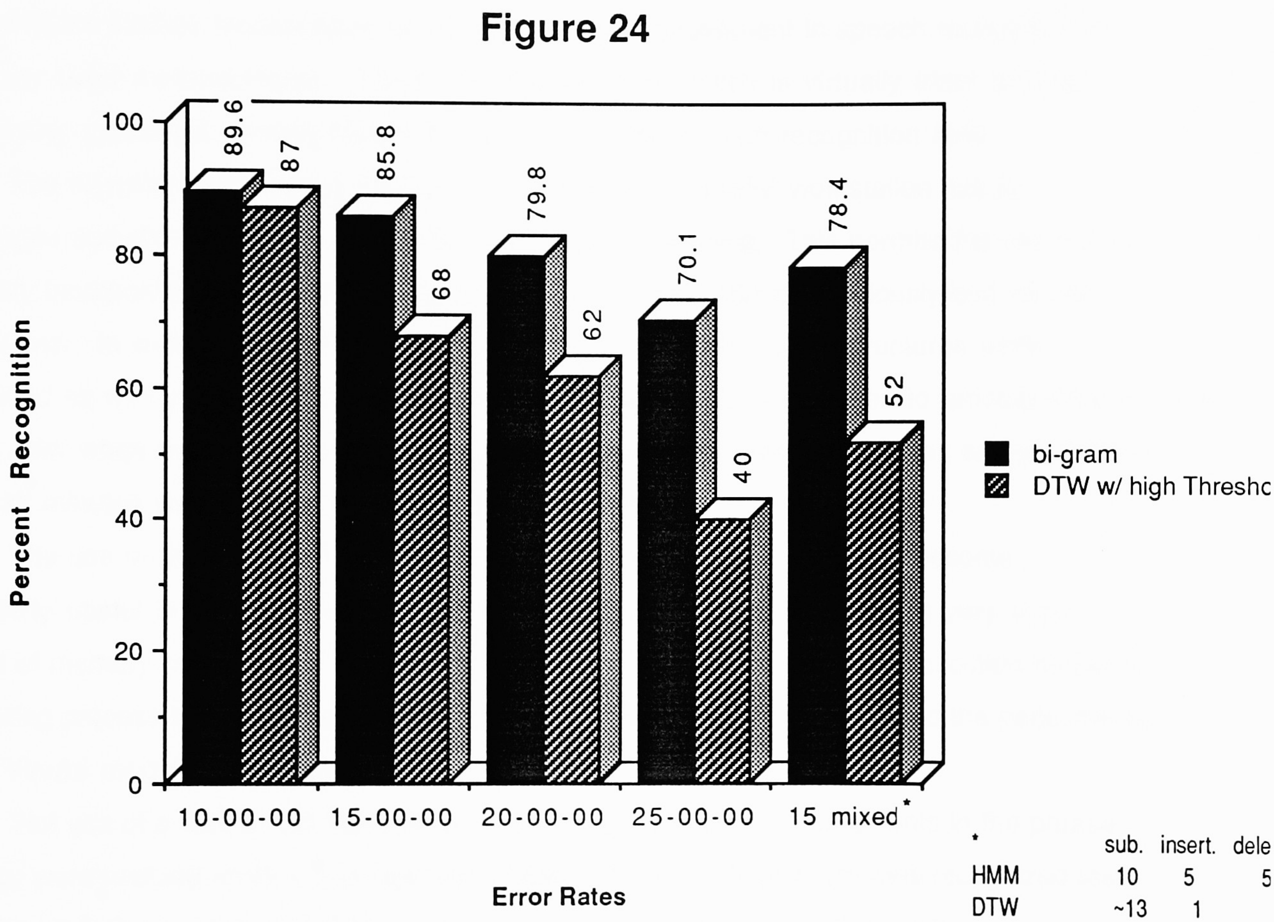


Results Showing the Benefit of the Word Lattice



** overall results from all tests conducted*

Comparison of HMM and DTW Results



*	sub.	insert.	dele
HMM	10	5	5
DTW	~13	1	

From these two viewpoints, the HMM method has proved more effective and efficient in the hypothesizing of words than the DTW method.

4. Conclusions

Hidden Markov Models have provided a marked improvement in speech recognition over previously used methodologies. Their inherent structure, which is virtually ideal for handling time-varying processes, makes HMMs a forerunner in the speech recognition field.

The implementation of the Hidden Markov Models on a LISP workstation has its advantages and disadvantages. LISP allows for rapid prototyping. This permits the programmer to quickly incorporate new ideas and approaches. However, LISP is notoriously bad for number calculations. In order to model the entire cockpit lexicon, many array structures were maintained as well as large lists. Even though LISP was primarily designed to process lists, it is very slow when the lists become lengthy. With no grammar introduced to the cockpit lexicon, nearly 12 minutes was required to process a single phoneme.

The use of the Explorer[3] provided an ability to use Flavors. This may become particularly useful in the integration of the entire speech project. However, a very large amount of memory was required to implement the HMM technology. Garbage collection had to be an ongoing process so that virtual memory space was not exceeded. This slowed the performance of the Viterbi module significantly.

The use of a little knowledge can go a long way. Dramatic improvements in the phrase accuracy were realized when a grammar was added. The percentage of phrases recognized rose more than 2.5 times when knowledge was introduced. But, there was little gain in using bi-gram over word pair grammar. This can be attributed to the limited data provided about the manner in which the words are used and the overall static nature of the HMM. A more complete index of the situations certain word phrases are used in would definitely increase the accuracy.

The HMM method performed very well when there were no insertion or deletion errors permitted for a large lexicon. A sharp decrease in the performance was noted when these errors were introduced and the substitution error rate was above 10%. The confusion created by these insertion and deletions grew as this study progressed from the digit lexicon to the cockpit lexicon. One could surmise that this trend would likely continue with an increase in the lexicon

size. So, the HMM is sensitive to insertion and deletion errors. One possible method for de-sensitizing the model would be to segment the phoneme string. This would give the system a better idea of where words start and finish.

The lexical search space was a key factor in the results obtained. Currently there exists no algorithmic solution to determine the proper size of the search space. As an alternative, a threshold could be used to prune unwanted candidates, but the problem is then at what value should the threshold be set. Lee points out that an adaptive beam width algorithm is required so that the search space can be reduced while maintaining good performance [LEE90]. Such a procedure does not yet exist.

Continued research in the application of HMMs should be conducted to provide faster and more accurate results. One solution would be to hard code the Viterbi into the system hardware. The extension of this study to incorporate the use of phonological rules in the creation of errorful phoneme strings would result in the generation of more realistic error in the phoneme string. Additionally, the results of the HMM could be improved by the use of prosodic information for the segmentation of the phoneme string. Another area of research would be the integration of HMMs and Neural Networks into a Viterbi net.

5. Bibliography

[ADAM86] Adams, D.A. and Bisiani, R., "The Carnegie-Mellon University Distributed Speech Recognition System", *Speech Technology,* March, 1986.

[BAKE75] Baker, J. K., "The DRAGON System - An Overview", *IEEE Transactions on Acoustics, Speech, and Signal Processing,* Vol. 23, February, 1975.

[BAKI76] Bakis, R., "Continuous Speech Recognition via Centisecond Acoustic States", *91st Meeting of Acoustical Society of America,* April, 1976.

[BAUM72] Baum, L. E., "An Inequality and Associated Maximization Technique in Statistical Estimation of Probabilistic Functions of Markov Processes", *Inequalities,* Vol. 3, 1972.

[BROW87] Brown, P., "The Acoustic-Modeling Problem in Automatic Speech Recognition", PhD Thesis, Computer Science Department, Carnegie-Mellon University, May 1987.

[COLE80] Cole, R. A., Rudnicky, A. I., Zue, V. W., and Reddy, D. R., Perception and Production of Fluent Speech, Lawrence Erlbaum Associates, Hillsdale, N.J., 1980.

[HATO84] Haton, J.P., New Systems and Architectures for Automatic Speech Recognition and Synthesis, Dordrecht, Reidel, Netherlands, 1984.

[HILL87] Hillenbrand, J. and Gayvert, R. T., "Speaker-Independent Vowel Classification Based on Fundamental Frequencies and Formant Frequencies", *Journal of Acoustical Society of America,* Spring 1987, Vol. 81 (suppl 1), s93(A).

[HOWA71] Howard, R. A., Dynamic Probabilistic Systems, John Wiley and Sons, Inc., New York, 1971.

[JELI76] Jelinek, F., "Continuous Speech Recognition by Statistical Methods", *Proceedings IEEE,* Vol. 64, 1976.

[LEE89] Lee, K. F., Automatic Speech Recognition - The Development of the SPHINX System, Kluwer Academic Publishers, Norwell, Mass., 1989.

[LEE90] Lee, C. H., Rabiner, L. F., Pieraccini, R., and Wilpon, J. G., "Acoustic Modeling of Subword Units for Speech Recognition", *Proceedings IEEE,* February, 1990.

[LEVI83] Levinson, S. E., Rabiner, L. R., and Sondhi, M. M., "An Introduction to the Application of the Theory of Probabilistic Functions on a Markov Process to Automatic Speech Recognition", *The Bell System Technical Journal,* Vol. 62, April, 1983.

[LEVI90] Levinson, S. E., Ljolje, A., and Miller, L. G., "Continuous Speech Recognition from a Phonetic Transcription", *Proceedings IEEE,* February, 1990.

[LOWE77] Lowerre, B. T., "Dynamic Speaker Adaptation in the HARPY Speech Recognition System", *IEEE International Conference Record on Acoustics, Speech and Signal Processing,* May 9-11, 1977.

[MARK07] Markov, A. A., "Extension of the Limit Theorems of Probability Theory to a Sum of Variables Connected in a Chain", The Notes of the Imperial Academy of Sciences of St. Petersburg, VIII Series, 1907.

[PORI88] Poritz, A. B., "Hidden Markov Models: A Guided Tour", *Proceedings ICASSP,* Tokyo, 1986.

[RABI86] Rabiner, L. R. and Juang, B. H., "An Introduction to Hidden Markov Models", *IEE ASSP Magazine*, January, 1986.

{SELM89] Selman, R. T., "Word Hypothesis of Undifferentiated, Errorful Phoneme Strings", MS Thesis, Department of Computer Science, Rochester Institute of Technology, May, 1989.

[SHEP80] Shepard, R., "Multidimensional Scaling, Tree-Fitting, and Clustering", *Science*, Vol. 210, October, 1980.

[STEE84] Steele Jr., G. L., COMMON LISP: The Language, Digital Press, Hanover, Ma., 1984.

[THOM87] Thompson, H. S. and Laver, J. D., "The Alves Speech Demonstrator - Architecture, Methodology to Date", *Proceedings of Speech Technology*, 1987.

[VITE67] Viterbi, A. J., "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm", *IEEE Transactions on Information Theory*, Vol. 13, April, 1967.

[WAIB86] Waibel, A. H., "Prosody and Speech Recognition", PhD Thesis, Computer Science Department, Carnegie-Mellon University, October, 1986.

Appendix A

Phoneme Confusion Probabilities

	er	aa	ae	ah	ao	ax	eh	ih	ix	iy	ow	uh	uw	l	r	y	w
er	–	5.25	8.46	9.40	5.40	8.06	10.00	6.75	6.42	2.98	5.08	9.55	5.97	3.40	7.09	2.54	3.66
aa	3.14	–	5.14	33.93	10.08	16.97	2.35	1.45	2.54	0.84	7.77	4.97	2.79	3.57	1.69	0.78	2.00
ae	8.46	8.58	–	12.78	4.29	10.43	17.34	6.03	7.23	2.74	4.05	4.76	2.93	2.83	3.09	2.17	2.30
ah	4.00	24.14	5.44	–	9.79	24.14	2.75	1.57	2.76	0.82	6.58	7.10	3.08	3.29	2.01	0.82	1.70
ao	3.03	9.44	2.41	12.89	–	9.17	1.90	1.41	2.41	0.85	15.90	19.08	9.26	5.78	2.33	0.90	3.23
ax	4.53	15.93	5.86	31.86	9.19	–	3.26	1.94	4.78	1.05	2.58	7.24	3.61	3.08	2.22	0.88	1.99
eh	8.30	3.26	14.40	5.37	2.82	4.82	–	30.02	3.92	6.16	2.66	4.20	2.83	2.02	4.15	3.21	1.88
ih	5.84	2.09	5.22	3.19	2.17	2.99	31.34	–	7.01	19.64	2.13	3.47	2.71	1.84	2.96	5.45	1.94
ix	7.94	5.26	8.95	8.03	5.33	10.52	5.84	10.02	–	5.01	5.69	7.51	5.13	3.45	4.78	3.19	3.34
iy	4.90	2.30	4.49	3.16	2.49	3.07	12.19	37.21	6.64	–	2.49	3.65	3.25	2.15	3.03	6.20	2.79
ow	3.69	9.42	2.94	11.22	20.58	3.34	2.33	1.79	3.34	1.10	–	14.52	10.04	7.26	2.87	0.99	4.57
uh	4.60	3.99	2.29	8.02	16.36	6.20	2.44	1.93	2.92	1.07	9.63	–	28.21	4.09	2.48	0.96	4.81
uw	3.90	3.05	1.91	4.73	10.79	4.19	2.23	2.05	2.71	1.29	9.04	38.33	–	5.42	3.37	1.29	5.70
l	1.75	3.07	1.46	3.98	5.31	2.82	1.25	1.09	1.44	0.67	5.15	4.38	4.27	–	7.42	8.42	6.04
r	3.74	1.49	1.63	2.49	2.19	2.09	2.64	1.80	2.04	0.98	2.09	2.72	2.72	7.60	–	6.75	8.16
y	1.56	0.80	1.33	1.18	0.98	0.96	2.37	3.86	1.58	2.32	0.83	1.23	1.21	10.03	7.85	–	8.48
w	1.87	1.70	1.17	2.05	2.95	1.81	1.16	1.14	1.38	0.87	3.22	5.11	4.46	6.00	7.90	7.07	–
dx	–	–	–	–	–	–	–	–	–	–	–	–	–	2.86	2.86	3.72	3.52
ng	–	–	–	–	–	–	–	–	–	–	–	–	–	5.53	7.37	4.79	5.53
b	–	–	–	–	–	–	–	–	–	–	–	–	–	4.40	6.01	4.93	7.25
t	–	–	–	–	–	–	–	–	–	–	–	–	–	1.76	1.93	1.79	1.97
d	–	–	–	–	–	–	–	–	–	–	–	–	–	2.86	2.86	3.72	3.52
k	–	–	–	–	–	–	–	–	–	–	–	–	–	1.85	2.07	1.87	2.07
g	–	–	–	–	–	–	–	–	–	–	–	–	–	2.95	3.02	3.65	3.65
m	–	–	–	–	–	–	–	–	–	–	–	–	–	4.88	6.03	4.26	4.95
n	–	–	–	–	–	–	–	–	–	–	–	–	–	7.46	8.47	5.60	6.15
p	–	–	–	–	–	–	–	–	–	–	–	–	–	1.79	2.02	1.88	2.04
hh	–	–	–	–	–	–	–	–	–	–	–	–	–	2.07	2.40	2.25	2.52
f	–	–	–	–	–	–	–	–	–	–	–	–	–	2.22	2.61	2.30	2.66
v	–	–	–	–	–	–	–	–	–	–	–	–	–	3.05	3.64	3.44	4.21
th	–	–	–	–	–	–	–	–	–	–	–	–	–	1.87	2.18	1.95	2.24
zh	–	–	–	–	–	–	–	–	–	–	–	–	–	3.10	3.16	3.55	3.62
dh	–	–	–	–	–	–	–	–	–	–	–	–	–	2.97	3.40	3.50	4.04
s	–	–	–	–	–	–	–	–	–	–	–	–	–	2.02	2.28	2.14	2.41
z	–	–	–	–	–	–	–	–	–	–	–	–	–	2.77	2.97	3.20	3.31
sh	–	–	–	–	–	–	–	–	–	–	–	–	–	2.23	2.44	2.35	2.57
ch	–	–	–	–	–	–	–	–	–	–	–	–	–	1.85	2.05	1.92	2.11
jh	–	–	–	–	–	–	–	–	–	–	–	–	–	3.96	4.58	5.48	6.67

dx	ng	b	t	d	k	g	m	n	p	hh	f	v	th	zh	dh	s	z	sh	ch	jh
.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2.54	3.37	3.13	0.97	2.54	1.01	2.11	2.58	4.17	1.09	1.42	1.45	2.19	1.35	1.48	2.22	1.24	1.67	1.00	0.99	2.97
2.60	4.60	4.38	1.09	2.60	1.16	2.22	3.26	4.85	1.26	1.69	1.74	2.68	1.60	1.55	2.60	1.44	1.83	1.11	1.12	3.52
3.94	3.48	4.17	1.18	3.94	1.22	3.11	2.68	3.73	1.36	1.85	1.78	2.94	1.67	2.03	3.11	1.57	2.29	1.25	1.22	4.90
3.10	3.34	5.11	1.08	3.10	1.12	2.60	2.60	3.41	1.23	1.72	1.72	3.00	1.60	1.72	3.00	1.47	1.98	1.14	1.12	4.97
-	1.86	3.29	1.16	32.87	1.15	10.96	1.66	1.88	1.26	1.79	1.63	3.18	1.62	3.87	4.48	1.69	3.94	1.38	1.23	6.16
2.71	-	4.96	1.84	2.71	2.02	2.52	15.13	14.37	2.18	2.66	2.99	3.42	2.61	1.94	3.06	2.21	2.32	1.71	1.82	3.59
4.11	4.25	-	1.94	4.11	2.02	3.91	3.68	3.63	2.28	3.63	3.57	9.13	3.24	2.65	6.85	2.90	3.74	2.04	2.02	7.70
1.87	2.03	2.50	-	1.87	12.20	1.98	2.11	1.82	12.69	4.96	5.12	2.86	6.10	1.92	2.52	5.66	2.40	5.38	14.42	2.14
32.87	1.86	3.29	1.16	-	1.15	10.96	1.66	1.88	1.26	1.79	1.63	3.18	1.62	3.87	4.48	1.69	3.94	1.38	1.23	6.16
1.86	2.26	2.63	12.32	1.86	-	1.95	2.41	1.99	20.02	4.85	5.93	2.89	6.41	1.84	2.48	4.78	2.29	3.91	7.28	2.19
13.59	2.15	3.88	1.53	13.59	1.49	-	1.94	2.11	1.65	2.45	2.13	4.37	2.16	6.80	6.80	2.33	7.41	1.91	1.65	6.80
2.77	17.47	4.95	2.21	2.79	2.50	2.63	-	10.06	2.68	3.07	3.69	3.73	3.16	2.10	3.22	2.57	2.51	2.00	2.16	3.61
2.96	15.68	4.61	1.79	2.99	1.95	2.70	9.50	-	2.08	2.53	2.77	3.34	2.49	2.08	3.10	2.16	2.41	1.72	1.78	3.69
1.82	2.36	2.66	11.50	1.83	17.97	1.94	2.32	1.90	-	5.75	7.19	3.03	8.22	1.82	2.54	5.53	2.32	1.82	7.57	2.18
2.32	2.36	3.75	3.98	2.32	3.86	2.55	2.36	2.05	5.09	-	9.10	5.09	14.15	2.27	3.86	10.61	3.35	4.11	4.55	3.03
2.20	2.80	3.89	4.33	2.22	4.97	2.33	2.98	2.38	6.71	9.59	-	4.48	16.78	2.05	3.40	5.97	2.77	3.32	4.20	2.86
3.94	2.91	9.04	2.20	3.94	2.20	4.36	2.74	2.60	2.57	4.88	4.07	-	4.00	3.13	12.20	3.94	5.31	2.54	2.35	6.78
2.00	2.21	3.21	4.69	2.00	4.87	2.16	2.32	1.93	6.96	13.53	15.23	3.99	-	1.93	3.12	7.86	2.65	3.75	4.87	2.49
7.04	2.47	3.98	2.22	7.18	2.10	10.17	2.32	2.42	2.32	3.27	2.79	4.69	2.91	-	6.20	3.36	10.46	3.00	2.44	5.23
5.33	2.49	6.51	1.86	5.33	1.82	6.51	2.28	2.32	2.07	3.55	2.97	11.72	3.01	3.97	-	3.21	7.82	2.28	2.02	9.02
2.41	2.19	3.35	5.09	2.43	4.25	2.71	2.21	1.96	5.48	11.87	6.33	4.59	9.19	2.61	3.90	-	3.70	7.49	6.47	2.91
5.82	2.35	4.41	2.21	5.82	2.08	8.82	2.21	2.24	2.35	3.83	3.00	6.33	3.16	8.32	9.70	3.78	-	3.03	2.47	5.82
2.73	2.34	3.25	6.66	2.75	4.79	3.07	2.37	2.16	5.53	6.34	4.85	4.09	6.04	3.22	3.81	10.34	4.09	-	8.93	3.07
2.05	2.07	2.68	14.89	2.05	7.44	2.21	2.13	1.86	8.62	5.85	5.12	3.15	6.55	2.18	2.82	7.44	2.78	7.80	-	2.37
7.29	2.92	7.53	1.58	7.29	1.60	6.67	2.54	2.71	1.77	2.78	2.48	6.48	2.38	3.38	9.34	2.38	4.67	1.82	1.69	-

Appendix B-1

Digit Grammar

(zero zero)
(zero zero)
(zero zero)
(zero zero)
(zero two)
(zero four)
(zero six)
(zero eight)
(one one)
(one one)
(one one)
(one one)
(one three)
(one five)
(one seven)
(one nine)
(two zero)
(two two)
(two two)
(two two)
(two two)
(two four)
(two six)
(two eight)
(three one)
(three three)
(three three)
(three three)
(three three)
(three five)
(three seven)
(three nine)
(three three)
(four zero)
(four two)
(four four)
(four four)
(four four)
(four four)
(four six)
(four eight)
(five one)
(five three)

(five five)
(five five)
(five five)
(five five)
(five seven)
(five nine)
(six zero)
(six two)
(six four)
(six six)
(six six)
(six six)
(six six)
(six eight)
(seven one)
(seven three)
(seven five)
(seven seven)
(seven seven)
(seven seven)
(seven seven)
(seven nine)
(eight zero)
(eight two)
(eight four)
(eight six)
(eight eight)
(eight eight)
(eight eight)
(eight eight)
(nine one)
(nine three)
(nine five)
(nine seven)
(nine nine)
(nine nine)
(nine nine)
(nine nine)

Appendix B-2

Cockpit Grammar

(POSITION OF RENDEZVOUS)
(STATE DATA)
(SPEAK)
(REQUEST RENDEZVOUS DATA)
(SHOW ME)
(CONTINUE)
(WHERE IS THE ATTACK ELEMENT)
(STRIKE FLIGHT)
(RENDEZVOUS POINT)
(RANGE BEARING)
(DISPLAY HORIZONTAL SITUATION OF STRIKERS)
(DIRECTION)
(ALLIED STATUS)
(SPEAK TO ME)
(VECTOR BOMBERS)
(T-O-T STATUS)
(RANGE AND BEARING WINGMAN)
(WHERE ARE THEY)
(WHERE IS THE MUD SEATER)
(SHOW IT)
(GO)
(SHOW RENDEZVOUS DATA)
(SHOW ME ATTACK FLIGHT)
(ACCESS RENDEZVOUS DATA)
(BEARING)
(SHOW)
(THREAT STATUS)
(RADAR SORT)
(IS IT IN AN ACTIVE MODE)
(THREAT DATA)
(IS IT A THREAT)
(GO)
(EVASIVE COURSE)
(AM I TARGETED)
(DISPLAY INFORMATION)
(THREAT STATS)
(DEFEAT)
(MORE DATA)
(ANALYZE)
(GIVE ME THE THREAT DATA)
(PRESENT THE DATA)
(GIVE ME MORE INFORMATION ON THE THREAT)
(THREAT RING)

(GO DATA)
(DESCRIBE THREAT)
(GIVE ME A THREAT RING)
(IS THE SITE ACTIVE)
(GIVE ME THE DATA)
(THREAT LOCKED ON TO ME)
(GO AHEAD)
(STATUS TEN)
(SAM ZONE)
(SHOW ME THE THREAT)
(SAY COUNTERMEASURES OPTIONS)
(WHAT ARE MY OPTIONS)
(IN RANGE)
(DISPLAY JAMMING OPTIONS)
(EMPLOY COUNTERMEASURES)
(JAM TEN THREE SIXTY TWENTY)
(COUNTERMEASURE STATUS)
(DEFENSIVE SYSTEMS CHECK)
(ARMS)
(COUNTER TEN)
(BIT CHECK)
(S-A TEN JAM)
(SUGGEST COUNTERMEASURES)
(BIT E-C-M)
(JAM)
(COUNTER OPTIONS)
(CHAFF AND FLARES)
(E-C-M CHAFF FLARES)
(CHECK E-C-M)
(GO DEFENSIVE)
(PLOT ALTERNATE COURSE)
(GO COUNTERMEASURES)
(ARE COUNTERMEASURES UP)
(JAMMERS ON)
(SUGGEST ALTERNATIVE ROUTE)
(GIVE ME OPTIONS)
(DISPLAY COUNTERMEASURES)
(JAM J BAND)
(BUZZERS ON)
(GIVE ME JAMMING AND CHAFF)
(MAX JAM)
(ARM EXPENDABLES)
(CONFIRM E-C-M)

(CHAFF AND FLARES ARM)
(CHAFF TO PROGRAM)
(COUNTERMEASURES TEN)
(POD TO PROGRAM)
(JAM THREAT)
(DISPENSE E-C-M)
(ACTIVATE SYSTEM)
(COUNTER THE THREAT)
(ACTIVATE E-C-M)
(E-C-M ARM)
(GIVE ME E-C-M CHAFF AND FLARES)
(JAM THE SIGNAL)
(E-C-M SHIRK IN RANGE)
(MULTIPLE CHAFF)
(JAM TEN THREE SIXTY TWENTY)
(GIVE ME CHAFF)
(JAM RADAR)
(SALVO BURST)
(ACTIVATE COUNTERMEASURES)
(JAM IT)
(EVALUATE THREAT AND DEFEAT)
(RANGE AND BEARING OF WINGMAN)
(SHOW ME)
(SAY DATA)
(UPDATE THE RENDEZVOUS)
(WHERE ARE THEY)
(GODS EYE)
(PROCEED)
(GIVE ME THE DATA)
(EXPAND THE GODS EYE)
(RANGE AND BEARING OF STRIKERS)
(REQUEST FIGHTER POSITION)
(GO)
(DISPLAY)
(DISPLAY RENDEZVOUS)
(LOCATION)
(SPEAK)
(GO AHEAD)
(RENDEZVOUS DATA)
(UPDATE)
(ACCOMPLISH FENCE CHECK)
(FLIGHT FENCE CHECK)
(OK)

(GO FENCE CHECK)
(FENCE CHECK GO)
(FENCE CHECK)
(PERFORM FENCE CHECK)
(GO GREEN)
(FENCE CHECK NOW)
(ARM SYSTEMS)
(ACCOMPLISH)
(ARM ME UP)
(GREEN IT UP)
(DO IT)
(YES)
(DO THE FENCE CHECK)
(CLEAR)
(CHECK THE TERRAIN FOLLOWING GEAR)
(GIVE ME T-F FOR TWO HUNDRED FEET)
(PERFORM BIT CHECK)
(T-F-R STATUS CHECK)
(BIT CHECK T-F)
(GIVE ME STATUS OF TERRAIN FOLLOWING)
(CHECK THE RADAR)
(SHOW T-F-R STATUS)
(HOWS THE TERRAIN FOLLOWING EQUIPMENT)
(T-F-R STATUS ON)
(TERRAIN FOLLOWING CHECK)
(TERRAIN FOLLOWING)
(GIVE ME TERRAIN FOLLOWING RADAR)
(T-F-R)
(DISPLAY THE STATUS OF THE TERRAIN GEAR)
(SELECT TERRAIN FOLLOWING RADAR)
(SAY SYSTEM STATE)
(SYSTEMS CHECK)
(T-F STATUS)
(ALTITUDE SAFE)
(CHECK STATUS T-F)
(PREPARE INGRESS)
(STATUS OF TERRAIN FOLLOWING)
(LOW LEVEL NAVIGATION STATUS)
(TERRAIN FOLLOWING DISPLAY)
(CONFIRM GREEN)
(RADAR ON WORKING OK)
(LETS GO AUTO)
(GIVE ME AUTO)

(T-F GO MANUAL)
(AUTO T-F)
(SELECT MANUAL)
(AUTO T-F-R)
(AUTO MANUAL)
(SELECT AUTO T-F)
(TERRAIN FOLLOWING AUTO)
(ENGAGE AUTO)
(T-F AUTO)
(TERRAIN FOLLOWING)
(GO AUTO)
(WHATS THE THREAT)
(SPEAK)
(GO)
(SHOW THREAT)
(EXPAND THREAT)
(SAY THREAT)
(THREAT INFO)
(PRIORITIZE)
(GO WITH THREAT DATA)
(STATE THREAT DATA)
(MORE DATA)
(DISPLAY PRIORITIZED)
(IDENTIFICATION OF THREAT)
(SHOW ME)
(THREAT DATA)
(THREAT DATA AVAILABLE)
(UPDATE)
(THREAT INFORMATION)
(THREAT DATA CONTINUE)
(GIVE ME THE DATA)
(SHOW ME KILLER THREATS)
(SHOW ME THE HIGHEST PRIORITY THREAT OR THE NEW THREAT)
(DISPLAY THE THREAT)
(THREAT STATUS)
(SELECT ORDNANCE)
(COUNTER)
(REQUEST AIR TO AIR MODE)
(GIVE ME THE RADAR MISSILE)
(SELECT BEST WEAPON)
(TARGET THE THREAT)
(MISSILE ARM)
(AVOID)

(THREAT NON THREAT)
(WEAPONS DISPLAY)
(TARGET HELO)
(TARGET LONG RANGE)
(TARGET RADAR)
(ARM AIR TO AIR GUNS)
(BRING UP THE ARMAMENT)
(ARM LONG RANGE MISSILE)
(ARM THEM OUT)
(AIM SEVEN ARM)
(GREEN THEM UP)
(AIR TO AIR WEAPONS)
(SPARROW)
(TARGET THAT THREAT ARM)
(ARM SIDEWINDER)
(SELECT WEAPON)
(GIVE ME RADAR)
(L-R-S THREAT)
(OFFENSE OR DEFENSE)
(DISPLAY BEST OPTION)
(GIVE MAX RANGE)
(LOCK HIM UP)
(ARM AIR TO AIR MISSILES)
(LOCK)
(SHORT READY)
(SHOW ME SAM)
(OFF TRACK ONLY)
(LOCK THREAT)
(STATE RANGE)
(LOCK ON TARGET TEN LEFT AT EIGHTEEN)
(LOCK ON NEWEST AIR TO AIR THREAT)
(GIVE ME A LOCK ON)
(STEERING CUE)
(LOCK ON AIR TO AIR THREAT)
(IN RANGE)
(LOCK ON ARM MISSILES)
(MISSILE LOCK)
(BORE SITE LIMA)
(GIVE ME MISSILE TARGET DESIGNATION ON WHEN TO TAKE THE SHOT)
(ENGAGE)
(REQUEST RANGE INFORMATION)
(LOCK ON THREAT)
(WHEN DO I)

(LOCK THREAT ONE TWO)
(TARGET THREAT)
(WEAPONS LAUNCH CRITERIA)
(SHOOT CUE)
(DISPLAY OPTIMUM RANGE)
(GIVE MAX RANGE)
(SPEAK)
(DISPLAY THREATS)
(MORE INFORMATION)
(WHAT IS IT)
(COUNTERMEASURES AND ALTERNATE ROUTE)
(TELL ME MORE)
(SHOW RANGE E-C-M CHAFF)
(DISPLAY THREAT INFORMATION)
(HIGHLIGHT THE THREAT)
(JAM HIM)
(SHOW DATA)
(JAM)
(I WANT MORE THREAT DATA)
(SAY THREAT)
(S-A EIGHTEEN)
(DEFEND SAM EIGHTEEN)
(GO AHEAD)
(THREAT DEFEAT)
(WHERES THE THREAT)
(AM I TARGET)
(AM I TARGET)
(DISPLAY THREAT RADIUS)
(DISPLAY THREAT)
(THREAT)
(DISPLAY)
(SHOW THREAT)
(AM I IN TROUBLE)
(GIVE ME THE DATA)
(DISPLAY BEST OPTION)
(EXECUTE REROUTE)
(SHOW ME)
(PICTURE REROUTE)
(STATE REROUTE)
(WHAT IS IT)
(DISPLAY ROUTE)
(SELECT BEST ROUTE)
(SHOW SAFEST REROUTE)

(SHOW ME A NEW ROUTING)
(DESCRIBE REROUTE)
(SHOW ALTERNATE ROUTE)
(GIVE ME THE NEW ROUTE DATA)
(REQUEST ROUTE)
(WHERE IS IT)
(DISPLAY AN AVOIDANCE ROUTE)
(SPECIFY REROUTE)
(DISPLAY ALTERNATE ROUTE)
(DISPLAY NEW ROUTE)
(SHOW SAFEST REROUTE)
(I WANT TO SEE IT)
(NEW HEADING)
(REROUTE OPTIONS)
(DISPLAY REROUTE)
(DESCRIBE THREAT)
(REROUTE)
(PASS REROUTE INFO TO FORMATION)
(SHARE INFORMATION)
(TELL THE REST OF THE FLIGHT)
(UPDATE THE ROUTE FOR H-T-E)
(LETS GO)
(SELECT NEW ROUTE)
(PASS INFO TO FORMATION)
(ALTERNATE ROUTING GO)
(LOCK)
(PASS REROUTE TO FLIGHT)
(ROUTE RIGHT)
(EXPRESS)
(SEND HORIZONTAL SITUATION TO OTHER FLIGHT MEMBERS)
(ACTIVATE ALTERNATE ROUTE)
(SEND INFORMATION)
(ZAP DISPLAY)
(UPDATE WINGMAN ON NEW ROUTE)
(SELECT REROUTE)
(MAGNUM AND SABER)
(START NEW ROUTE)
(SEND DATA)
(I TAKE THE NEW ROUTE AND TELL THE FLIGHT)
(REROUTE FORTY FIVE RIGHT)
(TELL TWO)
(THREAT REROUTE)
(SEND FORMATION)

(ROUTE CHANGE DISPLAY NEW ROUTE)
(ACCEPT NEW COURSE)
(ZAP IT)
(ROUTE DATA TO FLIGHT)
(WERE GOING THREAT EVASION)
(PROGRAM REROUTE)
(SEND)
(SEND IT)
(NEGATIVE COURSE)
(THREAT EVASION RIGHT)
(ZAP)
(RADIO CALL)
(SEARCH HIGH)
(STANDBY)
(ARM UP LONG RANGE MISSILES)
(ARM)
(ARM AIR TO AIR)
(SEARCH)
(SWITCH AIR TO AIR)
(DOGFIGHT RADAR)
(SELECT AIR TO AIR)
(AIR THREAT LOCK)
(AIR TO AIR MODE)
(DISPLAY GROUND TO AIR THREATS)
(LOCK ON THREAT)
(AIR TO AIR)
(WEAPON SELECT AIR TO AIR)
(AIR INTERCEPT MODE)
(TARGET BOGEYS)
(GO SEARCH)
(THREAT AIR)
(RADAR ENTER TARGETS INTO TRACK FILE)
(RADAR AIR TO AIR MODE)
(GO MISSILES)
(GO AIR TO AIR MODE)
(GO AIR TO AIR)
(CONFIGURE AIR TO AIR)
(REMODE AIR TO AIR)
(MASTER ARM)
(SELECT MEDIUM ALTITUDE AIR TO AIR MISSILE)
(GO AIR TO AIR)
(GIVE ME A LONG RANGE)
(READY AIR TO AIR MISSILES)

(ARM AMRAAM)
(SHOW STORES)
(ARM MISSILES)
(ARM THEM UP)
(GIVE ME THE RADAR MISSILE)
(SELECT RADAR)
(REQUEST ORDNANCE)
(DATA LINK TO TWO THREE FOUR)
(LETS LOCK THEM UP ON RADAR)
(SPARROW)
(MISSILES ARMED)
(MISSILE ARM)
(SELECT AMRAAM)
(L-R-S)
(ARM LONG RANGE AIM)
(SELECT AIR TO AIR MISSILE)
(SELECT BEST WEAPON)
(LONG RANGE RADAR MISSILE)
(AIR TO AIR MISSILE STANDBY)
(SELECT M-R-M)
(HEAT SENSOR)
(DISPLAY HIGHEST PRIORITY THREATS)
(I-R TRACKING)
(HEAT TRACK ON)
(INFRA RED SIGNATURE)
(I-R)
(ACTIVATE I-R)
(GIVE ME PASSIVE DETECTION INFORMATION)
(GIVE ME AN I-R SEARCH)
(PASSIVE SEARCH)
(DISPLAY I-R INFORMATION)
(I-D)
(INFRA RED)
(THREAT ASSESSMENT)
(I-R SEARCH TRACK)
(ATTEMPT TRACK I-R-S-T)
(HIDE)
(EXPAND INFO AIR TO AIR THREAT)
(DISPLAY HEAT)
(HEAT)
(DISPLAY ALL AIR TO AIR THREATS)
(HOSTILE)
(CONTINUE I-D)

(KEEP UPDATING ME)
(ATTEMPT I-D)
(PARROT)
(CONTINUE ATTEMPTS)
(SHOW BEST SHOT)
(IS HE A THREAT)
(THREAT INFORMATION)
(DETAIL THREAT)
(FRIEND OR FOE)
(INTERROGATE)
(ANALYZE FRIEND OR FOE)
(I-R-S-T)
(INTERROGATE A-P-X)
(INTERROGATE TARGET TWENTY LEFT TWO HUNDRED MILES)
(TARGET STATUS)
(INTERROGATE THEM)
(IDENTIFY WHEN HOSTILE)
(CLOSE UP)
(TARGET I-D)
(CLOSER LOOK)
(INFORMATION)
(NARROW BAND)
(NARROW)
(NARROW SWEEP)
(GIVE ME DETAILS)
(SORT FORMATION)
(ZOOM IN THE THREAT)
(BREAK OUT)
(DEFINE FORMATION)
(ENLARGE)
(PIN POINT TARGETS)
(GO SPOTLIGHT)
(I-D ZOOM)
(GIVE ME A BLOWUP OF TARGET AREA)
(EXPAND)
(SORT)
(ZOOM)
(TRACK)
(TARGET FLIGHT PATH)
(TIME TO INTERCEPT)
(THREAT FLIGHT PATH ANALYSIS)
(AIRCRAFT SETTING)
(CLOSE LOOK)

(NUMBER AND TYPE OF THREAT)
(I-D BOGEYS)
(REQUEST SPECIFIC INFORMATION)
(SPEED AND DIRECTION)
(PRESENT HOSTILE VECTOR)
(TRACK PATH)
(TRACK ANALYSIS)
(KILL THOSE MOTHERS)
(POINT OF ENCOUNTER)
(TARGET DATA)
(TARGET TRACK)
(DEFINE GEOMETRY)
(PROJECT FUTURE TRACK)
(PROJECT THEIR FLIGHT PATH)
(RADAR)
(AIR ATTACK UP)
(GIVE ME TRACK WHILE SCAN)
(ATTEMPT RADAR TRACK)
(RADAR ON TARGET DATA)
(WHEN IN RANGE LOCK AND INFORM ME)
(TRACK THREAT)
(SELECT AIR TO AIR MISSILE)
(SELECT MANUAL)
(SHOW RADAR)
(ENTER TARGETS IN T-W-S FILE)
(RADAR SENSOR)
(TRACK WHILE SCAN)
(SELECT RADAR MISSILE)
(SELECT RADAR)
(RADAR ON)
(TARGET RADAR)
(SELECT AIR TO AIR RADAR)
(AIR TO AIR RADAR ON)
(GO RADAR)
(SORT)
(GIVE ME INTERCEPT PROFILES)
(DISPLAY INTERCEPT OPTIONS)
(DISPLAY INTERCEPT)
(INTERCEPT OPTION)
(OPTIONS)
(REQUEST INTERCEPT)
(TWO ATTACK RIGHT ELEMENT)
(SHOW BEST INTERCEPT)

(SAY OPTIONS)
(DISPLAY AIR TO AIR RADAR)
(ATTACK OPTIONS)
(BANDIT INTERCEPT VECTOR)
(SHOW ME THE INTERCEPT PROFILES)
(GIVE ME AN INTERCEPT PROFILE)
(INTERCEPT HOT)
(OPTIMUM)
(GIVE ME A SNAP VECTOR)
(VID TRACK)
(MORE INFORMATION)
(DEFINE INTERCEPT)
(TACTICAL OPTIONS)
(INTERCEPT)
(COLLISION)
(MISSION IMPACT)
(GO OPTION TWO)
(GO OPTION)
(EXECUTE TWO)
(NUMBER THE INTERCEPT)
(INTERCEPT TWO)
(CALL BY NUMBER)
(ATTACK OPTION TWO)
(ROUTE TWO)
(TARGET OPTION)
(GIVE ME ATTACK TWO)
(OPTION TWO)
(DISPLAY FAST INTERCEPT)
(CONFIRM FLIGHT ATTACK OPTION ONE)
(GIVE ME THE TIME TO INTERCEPT)
(OK PROGRAM US FOR PROFILE ONE)
(RELAY INFO)
(TRANSMIT DATA)
(DISPLAY GEOMETRY)
(TELL TWO TO LOCK BANDITS)
(TRANSMIT TACTIC)
(ADVISE WINGMAN)
(HOOK INTERCEPT)
(RELAY)
(SEND TO WINGMAN)
(NOTIFY WINGMAN WELL START THE INTERCEPT)
(SEND IT)
(DISPLAY SELECTED ATTACK GEOMETRY)

(OPTION PASS)
(SINGLE SIDED ATTACK)
(BEAM SEND)
(SHOW MY INTERCEPT)
(PASS TO WINGMAN)
(SHOW FLIGHT)
(WERE STRIPPED)
(DISPLAY PROFILE TWO FOR MY WINGMAN)
(TRANSMIT ATTACK OPTION)
(DATA LINK OPTION ONE)
(SHOW PATH)
(DATA LINK SCREEN)
(SEND MESSAGE)
(DISPLAY FORMATION)
(TIME TO TARGET)
(FORMATION STATUS)
(SHOW ME SABER POSITION)
(FRIENDLIES)
(DISPLAY STATUS OF ATTACK FLIGHT)
(UPDATE THE BOMBERS)
(STATUS OF STRIKERS)
(CONTINUE)
(ATTACKERS UPDATE)
(REQUEST POSITION OF STRIKE PACKAGE)
(T-O-T FOR STRIKERS)
(DISPLAY FRIENDLY POSITION)
(SNAKES POSITION)
(SAY STATUS OF SABER FORTY ONE)
(DISPLAY SABER)
(CONFIRM STRIKE LEAD)
(STRIKERS AND TARGETS RELATIVE POSITION)
(SHOW ME STRIKE FLIGHT)
(UPDATE)
(BOMBER STATUS)
(SHOW OVERALL SITUATION)
(POSITION AND STATUS OF STRIKE FLIGHT)
(UPDATE SABER FORTY ONE)
(SHOW FLIGHT STATUS)
(STRIKER POSITION)
(STRIKE UPDATE)
(BEST ATTACK VECTOR)
(INTERCEPT VECTORS)
(FLIGHT PATH)

(SHOW INTERCEPT)
(DISPLAY ATTACK VECTORS)
(MORE AID)
(DISPLAY COMMAND STEERING)
(SNAP TARGET)
(BOGEY DOPE)
(REQUEST VECTORS)
(SNAP VECTOR)
(AUTO INTERCEPT)
(UPDATE THE THREATS)
(VERBALIZE DATA)
(LOCK)
(VECTOR)
(VECTOR INTERCEPT)
(G-C-I)
(INTERCEPT TARGET)
(REQUEST VECTOR TO TARGET)
(TWO DEPLOY)
(LEAN NORTH EAST)
(TACTICS)
(SEND TWO INTERCEPT INSTRUCTIONS)
(ATTACK STEERING)
(FLY ATTACK FORMATION)
(TWO GO TACTICAL)
(TWO CLEARED OFF)
(VECTOR)
(CONFIRM SORT)
(DEPLOY RIGHT FOUR OCLOCK)
(DEPLOY LEFT)
(PINCH)
(ABREAST)
(GO TRAIL)
(GO OFFENSIVE)
(ATTACK FORMATION)
(RAM LEFT)
(ADVISE WINGMAN)
(VECTOR ZERO SIX ZERO DEPLOY)
(EXECUTE)
(GIVE TARGET ASSIGNMENT)
(DISPLAY TARGETED AIRCRAFT)
(DEPLOY)
(DISPLAY ASSIGNMENTS)
(GIVE ME A TARGET)

(ATTACK ASSIGNMENT)
(ASSIGN TARGETS)
(TARGET GO)
(REQUEST TARGET ASSIGNMENT)
(IDENTIFY TARGETS)
(SHOW ME WHICH ONE TWO SHOOT AT)
(SHOW TARGETS)
(DEPLOY WINGMAN)
(DISPLAY TARGETING)
(TARGETING)
(BEST TARGET)
(TWO TARGET LEADER RIGHT ELEMENT)
(HELP SORT)
(INTERCEPT INFORMATION)
(GIVE ME OPTIONS)
(GIVE ME TARGET PRIORITIES AND ASSIGNMENT)
(GOOD TARGETS)
(TARGET ASSIGNMENT SEND WINGMAN)
(TARGETING CONFIRMED)
(SEND TWO TARGET INFORMATION)
(ASSIGNMENT ACCEPT)
(ACCEPT SEND)
(ACCEPT)
(ATTACK CONFIRMED)
(PASS THE TARGET ASSIGNMENT TO TWO)
(NOTIFY WINGMAN TO TAKE TRAIL I-VE GOT LEAD OF FLIGHT ONE)
(RESORT)
(ACCEPTED AND PASS)
(CHECK TARGETED INFORMATION)
(ACCEPT SEND TO WINGMAN)
(ACCEPT OPTIONS)
(TARGET ASSIGNMENT ACCEPTED)
(ACCEPT OPTIONS)
(ACKNOWLEDGE)
(OK)
(I-M IN ON THE LEFT ONES)
(PASS TO THE FLIGHT)
(ASSIGNMENTS OK)
(TRAILER)
(LEAD SORTED)
(ZAP TWO)
(PASS DATA)
(TRANSMIT TARGET)

(I-M IN ON THE RIGHT ONES)
(TARGETING ACCEPTED)
(ACCEPTED)
(TARGETING SWITCH)
(RESORT FORGET TWO)
(OFFSET FIFTY FIVE RIGHT)
(TARGET SWITCH)
(SORT)
(WINGMAN INFO)
(MISSILE PARAMETERS)
(ATTACK STATUS)
(RETARGET)
(SWAP LEAD AND TWOS TARGET ASSIGNMENT)
(SWAP TARGETS)
(LOCK)
(SWAP)
(ATTACK TARGET TWO)
(INFORM SABER FORTY ONE)
(CHANGE SORT)
(SWING AND ZAP)
(SWITCH TARGET ASSIGNMENTS)
(TARGETS CONFIRMED)
(SHOW TWO)
(SWITCH)
(TARGETING CHANGE)
(TELL TWO TO TAKE THE LEADER)
(RETARGETING)
(RETARGET BANDITS)
(SPLIT)
(CHANGE TARGET ASSIGNMENTS)
(SWITCH TARGETS)
(CHECK TARGET DISPLAY)
(RESORT OPTION SELECTED)
(SEND MESSAGE)
(SWAP TARGETS)
(ACKNOWLEDGE)
(SWITCH THE TARGETS)
(TRANSMIT DATA)
(SEND TWO INFO)
(YOU-VE GOT THE ONES I-VE GOT THE TWOS)
(TRANSMIT TARGET)
(PASS CHANGES)
(RELAY)

(PASS THE PLAN)
(SWITCH AND ZAP)
(RELAY INFO)
(SEND MESSAGE)
(ADVISE WINGMAN)
(TARGET SWITCH WINGMAN)
(NOTIFY WINGMAN)
(PASS TO WINGMAN)
(SEND TWO INFO)
(RESORT OPTION SELECTED)
(LOCK ON RADAR)
(OPTIMUM WEAPON SELECTION)
(RADAR LOCK CALL IN RANGE)
(GREEN THEM UP)
(LOCK HIM UP)
(STATE WHEN IN RANGE)
(HEAT LOCK)
(MULTIPLE LAUNCH)
(M-R-M)
(GO)
(IN RANGE)
(REQUEST OPTIMUM SHOT)
(TARGET TRAILER AND DISPLAY WEAPONS PARAMETERS)
(ENGAGE)
(ENGAGE ALL TARGETS)
(GIVE ME SHOOT CUE AT OPTIMUM RANGE)
(CONFIRM WEAPONS)
(SAY FIRE IN RANGE)
(TARGET BOTH TARGETS)
(LOCK LEFT TARGET)
(CONTINUE UPDATE)
(MASTER ARM ON)
(TARGET TWO)
(SIMULTANEOUS FIRE)
(READY TWO)
(WEAPONS STATUS)
(GUN)
(SAY CLOSURE)
(S-R-M)
(SELECT TWO AIR TO AIR MISSILES)
(ARM THEM UP)
(UPDATE TARGETS)
(TARGETING)

(BEGIN CHAFF FLARES)
(COUNTERMEASURES)
(DEFEAT MIG THIRTY NINE)
(DEPLOY E-C-M)
(ACTIVATE CHAFF FLARES)
(RAIN)
(ARM UP THE MISSILES)
(E-C-M MAX)
(BUG OUT)
(DISPENSE CHAFF FLARES)
(CHAFF AND FLARES AUTO)
(INITIATE E-C-M PROGRAM FOR EVASION)
(DEPLOY CHAFF FLARES)
(DISPENSE)
(FLARES CHAFF)
(E-C-M DISPENSE)
(GO DEFENSIVE)
(SET UP CHAFF FLARES)
(OPTIMIZE EGRESS FOR MIG THIRTY NINE)
(GO COUNTERMEASURES)
(ARM CHAFF FLARES)
(FLARES NOW)
(DISPLAY SABER FLIGHT)
(SHOW H-S-D)
(GIVE WINGMAN REJOIN VECTOR)
(ALLIED STATUS)
(TWO WHERE ARE YOU)
(REQUEST POSITION ON TWO)
(REJOIN INFORMATION)
(VECTORS FOR JOIN UP)
(SAY REJOIN STATUS)
(DAMAGE CHECK)
(SNAP)
(SHOW THE WINGMAN)
(WINGMAN POSITION)
(WHERES TWO)
(DISPLAY WINGMAN)
(WING DATA)
(FUEL)
(REQUEST RENDEZVOUS INFORMATION)
(STATUS TWO)
(FORMATION STATUS)
(BLIND)

(SNAP FRIENDLY)
(WINGMAN STATUS REJOIN)
(SHOW WINGMAN)
(WINGMANS POSITION)
(RENDEZVOUS FIGHTERS)
(GIVE ME THREAT DATA AND TWOS POSITION)
(CHAFF FLARES)
(DISPLAY FUEL STATUS)
(INROUTE DATA)
(AIRCRAFT STATUS)
(SHOW STATUS)
(THREAT DATA)
(B-D-A)
(HOLD TIME AVAILABLE)
(BUG OUT)
(VECTOR HOME)
(DEFENSIVE STATUS)
(FENCE OUT)
(HOW WE DOING)
(REQUEST BATTLE DAMAGE CHECK)
(FUEL INFO)
(ASSESS)
(ARMAMENT STATUS)
(FUEL STATUS)
(E-C-M STATUS)
(OPS CHECK)
(BATTLE DAMAGE CHECK)
(PIGEONS)
(COUNTERMEASURES STATUS)
(CHECK GAS)
(FUEL AND DAMAGE CHECK)
(VECTOR R-T-B)
(VECTOR HOME PLATE)
(DAMAGE CHECK)
(SAY YES)
(ALTERNATE R-T-B)
(VECTORS TO ALTERNATE)
(H-S-D AND SCOPE)
(GO ALTERNATE)
(ROUTING TO ALTERNATE)
(ALTERNATE LOCATION)
(GIVE ME MY OPTIONS)
(SHOW ALTERNATE)

(WHERES THE ALTERNATE)
(DISPLAY AND ZAP)
(BEST PROFILE)
(SAY ALTERNATE)
(DISPLAY SUITABLE ALTERNATE)
(SPEAK)
(DISPLAY ALTERNATE BASES)
(MORE INFORMATION)
(ALTERNATE)
(VECTORS FOR ALTERNATE)
(SHOW OPTIONS)
(REQUEST VECTORS TO ALTERNATE)
(SELECT ALTERNATE TWO)
(SHOW)
(DISPLAY ALTERNATES)
(VECTOR TO ALTERNATE)
(ACCEPT HAHN)
(SELECT THE NUMBER TWO PROFILE)
(MAX RANGE MACH)
(GO ALTERNATE)
(VECTORS TO HAHN)
(ALTERNATE THREE)
(DISPLAY ROUTING TO ALTERNATE)
(RECOVER AT RHEIN MAIN)
(BEST ENDURANCE PROFILE)
(ROUTE TO HAHN)
(QUICK TURN AROUND)
(TIME AND LOCATION OF RENDEZVOUS)
(UPDATE THE ALLIED STATUS)
(WHERE ARE THE OTHER PLANES)
(REQUEST MY OPTIONS AND RENDEZVOUS DATA)
(DISPLAY HORIZONTAL SITUATION OF STRIKERS)
(POINT NUMBER ALLIED STATUS)
(RANGE AND BEARING WINGMAN)
(STATE DATA AND SHOW ME ATTACK FLIGHT)
(STATUS OF STRIKE FLIGHT)
(CAN I KILL HIM OR CAN I AVOID HIM)
(DESCRIBE THREAT AND DISPLAY THREAT RADIUS)
(GIVE ME MORE INFORMATION ON THE THREAT)
(IS IT IN AN ACTIVE MODE)
(PRESENT THE THREAT DATA)
(SHOW LETHAL RANGE)
(THREAT LOCKED ON TO ME)

(ARE COUNTERMEASURES UP)
(DISPLAY JAMMING OPTIONS)
(E-C-M AGAINST THE TEN)
(E-C-M CHAFF FLARES)
(JAM TEN THREE SIXTY TWENTY)
(SHOW ME THE BEST WAY TO DEFEAT)
(WHAT ARE MY OPTIONS)
(ACTIVATE E-C-M AGAINST THREAT)
(ARM EXPENDABLES)
(NOTIFY WINGMAN TO START THE INTERCEPT)
(WHEN IN RANGE LOCK AND INFORM ME)
(ARM TWO MISSILES GIVE ME IN RANGE ON BOTH)
(I TAKE THE NEW ROUTE AND TELL THE FLIGHT)
(TELL THE REST OF THE FLIGHT)
(DISPLAY SELECTED ATTACK GEOMETRY)
(LOCK ON TARGET ON THE NOSE THIRTY FIVE MILES)
(NAV MAP EXPAND ON L-R-S THREAT)
(WHAT KIND OF MISSILES DO I HAVE)
(BUZZERS ON)
(CHAFF FLARES SALVO TWO SECONDS)
(GIVE ME JAMMING AND CHAFF)
(INITIATE JAMMING)
(RANGE AND BEARING OF STRIKERS)
(REQUEST FIGHTER POSITION)
(WHERE IS EVERYBODY)
(ACCOMPLISH FENCE CHECK)
(ARM ME UP)
(DO THE FENCE CHECK)
(BIT CHECK TERRAIN FOLLOWING)
(CHECK T-F STATUS)
(CHECK THE TERRAIN FOLLOWING GEAR)
(GIVE ME T-F FOR TWO HUNDRED FEET)
(HOWS THE TERRAIN FOLLOWING EQUIPMENT)
(SHOW ME ANY HIGH DANGER THREATS AND AIR TO AIR THREATS)
(SHOW ME THE HIGH PRIORITY THREAT OR THE NEW THREAT)
(WHATS THE WORST THREAT BOX)
(AIR TO AIR ARMAMENT)
(ARM AIR TO AIR MISSILES)
(BRING UP THE AIR TO AIR MISSILES)
(GO AHEAD AND FIGHT HIM)
(LOCK HIM UP TO L-R-S THREAT)
(RADAR ENTER TRACK WHILE SCAN TARGET HELICOPTER)
(WHAT KIND OF MISSILES DO I HAVE)

(BORE SITE LIMA)
(DISPLAY OPTIMUM RANGE)
(DISPLAY WEAPONS PARAMETERS)
(GIVE ME MISSILE TARGET DESIGNATION ON WHEN TO TAKE THE SHOT)
(LOCK ON NEWEST AIR TO AIR THREAT)
(RADAR LOCK IS ON)
(AM I IN TROUBLE)
(DISPLAY AN AVOIDANCE ROUTE)
(FOLLOW THREAT EVASION PROFILE)
(GIVE ME THE DATA)
(I WANT TO KNOW WHAT THE DATA IS)
(PODS AND CHAFF DEFEAT EIGHTEEN)
(DISPLAY AN ALTERNATE ROUTE)
(GO THREAT EVASION ROUTE)
(I WANT TO SEE IT)
(MORE INFORMATION)
(ACCELERATE TO FIVE HUNDRED AND TEN)
(ALTER COURSE TO REROUTE NOTIFY WINGMAN AND PACKAGE)
(I TAKE THE NEW ROUTE AND TELL THE FLIGHT)
(NEW COURSE SELECTED DATA LINK)
(PASS REROUTE INFO TO FORMATION)
(SEND HORIZONTAL SITUATION TO OTHER FLIGHT MEMBERS)
(SEND NEW DATA TO WINGMAN FLIGHT)
(AIR TO AIR SELECT SPARROW ARM)
(EVALUATE THREAT INTERCEPT PROBABILITY)
(I-D AIRCRAFT TEN OCLOCK TWO HUNDRED MILES)
(RADAR ENTER TARGETS INTO TRACK FILE)
(SELECT COUNTERMEASURES FOR AIR TO AIR THREAT)
(DATA LINK TO TWO THREE FOUR)
(LETS LOCK THEM UP ON RADAR)
(EXPAND INFO AIR TO AIR THREAT)
(GIVE ME AN I-R SEARCH)
(GIVE ME PASSIVE DETECTION INFORMATION)
(ARE THEY ENEMY AIRPLANES)
(GIVE ME BOGEY STATUS)
(INTERROGATE TARGET TWENTY LEFT TWO HUNDRED MILES)
(GIVE ME A BLOWUP OF TARGET AREA)
(ATTEMPT TRACK ANALYSIS)
(DISPLAY TRACK HISTORY)
(KILL THOSE MOTHERS)
(PROJECT THEIR FLIGHT PATH)
(ARE BOGEYS A THREAT TO ME)
(ATTEMPT RADAR TRACK)

(DISPLAY RADAR PICTURE)
(GIVE ME TRACK WHILE SCAN)
(WHEN IN RANGE LOCK AND INFORM ME)
(RADIO CALL SIMULTANEOUSLY)
(RESORT OPTION SELECTED)
(YOU-VE GOT THE ONES I-VE GOT THE TWOS)
(DISPLAY WEAPONS PARAMETERS)
(ENGAGE ALL TARGETS)
(GIVE ME SHOOT CUE AT OPTIMUM RANGE)
(RADAR LOCK CALL IN RANGE)
(REQUEST OPTIMUM SHOT)
(SELECT TWO AIR TO AIR MISSILES)
(SHOOT THE SUCKERS)
(CHAFF AND FLARES AUTO)
(DEFEAT MIG THIRTY NINE)
(EXPEND CHAFF FLARE)
(INITIATE E-C-M PROGRAM FOR EVASION)
(OPTIMIZE EGRESS FOR MIG THIRTY NINE)
(SET UP CHAFF FLARES)
(GIVE ME THE BIG PICTURE)
(GIVE ME THREAT DATA AND TWOS POSITION)
(GIVE WINGMAN REJOIN VECTOR)
(SAY REJOIN STATUS)
(SHOW TWOS POSITION)
(SNAP VECTOR ONE TO TWO)
(TWO WHERE ARE YOU)
(VECTORS FOR JOIN UP WITH SABER)
(WHERE ARE MY ATTACKERS)
(ARE THERE ANY TANKERS AVAILABLE)
(FUEL AND DAMAGE CHECK)
(REQUEST BATTLE DAMAGE CHECK)
(SAY AIRPLANE STATUS)
(SYSTEM SELF TEST)
(SYSTEM STATUS CHECK)
(DISPLAY OPTIMUM CRUISE DATA)
(GIVE ME MY OPTIONS)
(H-S-D AND SCOPE)
(PLOT ROUTE TO ALTERNATE BASE)
(REQUEST HOMEPLATE INFORMATION)
(SET VECTOR NEAREST ALTERNATE)
(VECTORS TO THE ALTERNATE)
(WHERES THE ALTERNATE)
(BEST ENDURANCE PROFILE)

(DISPLAY ROUTING TO ALTERNATE)
(RECOVER AT RHEIN MAIN)
(VECTORS TO RHEIN MAIN)
(TURN ABOUT AND REPORT ACTION)
(TURN ABOUT AND REPORT ACTIVITY)
(REPORT ANY ADDITIONAL INFORMATION)
(SET AIMS)
(AIRPLANES ARE AIRBORNE)
(LOAD AMRAAMS)
(RETURN AS IS)
(ASSESS ASPECT)
(ATTACH FUEL LINE)
(DROP ATTEMPTED)
(ON AUTOMATIC)
(AUTO AVIONICS)
(HE IS AVOIDING US)
(REPORT BACK)
(BETTER BREAKOUT B-D AND B-V-R)
(CALCULATE C)
(HE IS CHIEF)
(CHOOSE WEAPON)
(CHOPPER CLEARED AREA)
(CLIMB UP)
(TARGET CLOSEST)
(COME ABOUT AND CLOSE FOR COMBAT)
(COMMENCE FIRE)
(COMMIT EVERYBODY)
(MISSION COMPLETE)
(REQUEST CONSENT)
(NOTIFY CONTROL)
(SWITCH CONVERSION AND COUNT)
(CROSS OVERALL)
(FIGHTER CROSSING)
(DESIGNATE TARGET AND DIRECT FIRE)
(DIVERT FIRE)
(DROP IMPLEMENT)
(FIGHTER ENGAGED)
(ENGAGEMENT ENVELOPE)
(ESCORT EVERYTHING TO RAMSTEIN)
(GIVE-ME FRONT)
(FULL SCALE)
(GET RANGE)
(LET GUYS GO)

(HEATER FOLLOWING)
(ATTACK HOSTILES)
(I HAD HIM)
(REPORT IRST)
(I-HAVE HIM)
(DESCRIBE JAMMER AND JAZZ UP)
(LIKE DATA)
(LINE DATA)
(LOAD DATA)
(ONE MAN)
(EXECUTE MANEUVER AND RETURN TO MEIN)
(MONITOR ON VISUAL)
(MOVE UP)
(NET HIM)
(CLIMB TO NINETY)
(SEPARATE OPERATIONAL NUMBERS)
(OPTIMAL RANGE)
(STORE OUR PERFORMANCE)
(PRIMARY OVERVIEW)
(REPORT POSIT THROUGH WINGMEN)
(HE IS A PINCER)
(WHICH P-K)
(WIDE PLATFORM)
(UPDATE PRESENTATION)
(PRESS RAID)
(UNKNOWN ROUTES)
(RETURN TO AREA)
(REPEAT REPEAT)
(RAW DATA)
(PUT TARGET ON VISUAL)
(READING VECTORS)
(REASSIGN VECTOR)
(VECTOR REASSIGNMENT)
(RECALCULATE DIRECTION)
(RECONFIGURE TRAILER)
(RECOVERY OPS)
(R-V R-W-R)
(SAMPLE THE SCALE ON SCANNERS)
(WEAPONS SCHEDULE)
(STEER RIGHT)
(SPARKLE SUB)
(WANT-TO T-F-T-A)
(TRANSFER VIEW)

(UPDATED SETTING)
(TWELVE SIXTY)
(WILCO OUT)
(I WILL SHIRK)
(YOUR V-SUB-C)
(WIDE WINDOW)
(NUMBER CONVERSION)
(MANEUVER REQUEST)
(REQUEST GRANTED)

Appendix C-1

Phrase Recognition Results for Digit Lexicon with no Grammar

Error rates are listed as substitution, insertion and deletion errors respectively.

The top of the data columns reflect the lexical search space parameters:

5	-	top 5 candidates retained
10	-	top 10 candidates retained
5 space	-	top 5 candidates retained and the correct phrase was found in the search space.
10 space	-	top 10 candidates retained and the correct phrase was found in the search space.
x	-	search space initially extended

	error-rates	5	5-space	10	10-space
1	05-00-00	96.700	98.000	99.300	100.000
2	05-05-05	85.300	90.000	84.000	90.000
3	05-10-10	76.700	82.700	75.300	84.000
4	05-15-15	70.700	78.700	73.300	85.300
5	05-20-20	58.700	70.000	58.000	70.000
6	10-00-00	94.700	96.000	98.000	100.000
7	10-05-05	76.000	76.600	90.000	92.000
8	10-10-10	70.700	78.600	68.000	74.600
9	10-15-15	61.300	70.000	65.300	73.300
10	10-20-20	54.000	64.600	65.300	73.900
11	15-00-00	90.600	92.000	100.000	100.000
12	15-05-05	77.300	80.600	81.300	85.300
13	15-10-10	73.300	79.400	72.600	77.300
14	15-15-15	56.000	62.600	65.300	72.600
15	15-20-20	57.300	66.000	56.000	65.300
16	20-00-00	89.300	90.700	94.600	100.000
17	20-05-05	80.000	83.400	80.000	87.300
18	20-10-10	72.000	72.700	69.300	79.300
19	20-15-15	62.700	68.000	58.700	67.300
20	20-20-20	52.000	60.000	52.600	62.700
21	25-00-00	90.600	92.700	96.000	100.000
22	25-05-05	70.000	74.700	80.600	90.000
23	25-10-10	65.300	68.000	66.700	75.400
24	25-15-15	53.900	60.700	67.300	74.700
25	25-20-20	52.000	60.000	57.300	67.900
26	30-00-00	80.000	82.700	92.600	100.000
27	30-05-05	70.000	74.000	76.700	84.600
28	30-10-10	66.000	72.700	71.300	77.900
29	30-15-15	50.600	56.700	52.700	67.300
30	30-20-20	52.600	57.300	50.700	65.300
31	35-00-00	76.000	82.000	94.000	99.300
32	35-05-05	68.000	74.000	75.300	84.700
33	35-10-10	58.000	62.600	64.000	76.600
34	35-15-15	55.300	64.000	58.000	66.000
35	35-20-20	42.000	48.600	44.600	59.300
36	40-00-00	76.000	78.000	90.700	98.700
37	40-05-05	63.300	70.000	72.000	88.000
38	40-10-10	56.700	64.000	60.000	70.000
39	40-15-15	44.000	54.000	61.300	74.000
40	40-20-20	38.700	47.300	48.000	64.000
41	45-00-00	69.300	74.600	86.000	97.900
42	45-05-05	55.300	64.700	67.300	85.300
43	45-10-10	55.300	60.000	59.300	72.700
44	45-15-15	42.700	52.000	54.000	67.300
45	45-20-20	38.000	44.000	48.000	59.400
46	50-00-00	64.600	73.300	82.700	96.700
47	50-05-05	56.000	63.300	67.300	84.000
48	50-10-10	52.000	56.700	50.000	66.700
49	50-15-15	43.300	48.000	42.700	66.700
50	50-20-20	34.700	45.300	48.000	60.700

	error-rates	x-5	x-5-space	x-10	x-10-space
1	05-00-00	97.300	98.000	100.000	100.000
2	05-05-05	82.000	86.000	85.300	93.300
3	05-10-10	74.000	80.700	78.700	90.700
4	05-15-15	66.700	76.000	52.000	76.000
5	05-20-20	53.300	69.300	58.700	76.000
6	10-00-00	94.700	94.700	98.700	100.000
7	10-05-05	85.300	88.700	78.700	88.000
8	10-10-10	70.700	75.300	74.700	85.300
9	10-15-15	62.700	68.700	66.700	78.700
10	10-20-20	55.300	66.000	50.700	68.000
11	15-00-00	94.700	95.300	96.000	100.000
12	15-05-05	75.300	78.000	77.300	86.700
13	15-10-10	70.700	75.300	65.300	85.300
14	15-15-15	62.700	70.700	61.300	76.000
15	15-20-20	56.700	68.000	58.700	76.000
16	20-00-00	86.000	88.000	97.300	100.000
17	20-05-05	76.700	81.300	81.300	86.700
18	20-10-10	72.700	77.300	69.300	80.000
19	20-15-15	53.900	62.700	61.300	69.300
20	20-20-20	46.700	51.300	57.300	72.000
21	25-00-00	86.000	87.300	93.300	100.000
22	25-05-05	72.700	76.700	77.300	89.300
23	25-10-10	64.000	70.700	68.000	81.300
24	25-15-15	56.700	64.700	58.700	74.700
25	25-20-20	55.300	62.700	56.000	74.700
26	30-00-00	88.700	90.000	92.000	97.300
27	30-05-05	68.700	74.000	72.000	80.000
28	30-10-10	65.300	70.700	64.000	81.300
29	30-15-15	57.300	63.300	66.700	80.000
30	30-20-20	56.000	61.300	50.700	68.000
31	35-00-00	80.700	86.700	90.700	98.700
32	35-05-05	67.300	76.000	78.700	92.000
33	35-10-10	61.300	69.300	70.700	82.700
34	35-15-15	52.000	60.000	60.000	78.700
35	35-20-20	52.700	58.700	52.000	72.000
36	40-00-00	74.700	78.700	92.000	98.700
37	40-05-05	64.700	72.000	66.700	81.300
38	40-10-10	546.000	62.700	64.000	81.300
39	40-15-15	57.300	66.700	60.000	78.700
40	40-20-20	48.700	56.700	49.300	73.300
41	45-00-00	72.000	81.300	88.000	98.700
42	45-05-05	54.700	68.700	66.700	82.700
43	45-10-10	48.700	62.700	70.700	80.000
44	45-15-15	55.300	63.300	52.000	69.300
45	45-20-20	49.300	60.700	34.700	60.000
46	50-00-00	72.700	82.700	78.700	96.000
47	50-05-05	51.300	65.300	66.700	84.000
48	50-10-10	42.000	52.700	62.700	80.000
49	50-15-15	46.000	58.000	50.700	69.300
50	50-20-20	38.700	53.300	52.000	70.700

Appendix C-2

Phrase Recognition Results for Digit Lexicon with Grammar

Error rates are listed as substitution, insertion and deletion errors respectively.

The top of the data columns reflect the lexical search space parameters:

5	-	top 5 candidates retained
1 0	-	top 10 candidates retained
5 space	-	top 5 candidates retained and the correct phrase was found in the search space.
10 space	-	top 10 candidates retained and the correct phrase was found in the search space.
x	-	search space initially extended
wp	-	word pair
bg	-	bi-gram

	error-rates	wp-5	wp-5-space	wp-10	wp-10-space
1	05-00-00	100.000	100.000	100.000	100.000
2	05-05-05	86.700	88.700	86.000	88.000
3	05-10-10	76.000	84.000	78.700	89.300
4	05-15-15	70.000	77.300	63.300	78.000
5	05-20-20	61.300	74.600	58.700	76.000
6	10-00-00	93.300	100.000	98.700	100.000
7	10-05-05	86.000	88.600	79.300	86.700
8	10-10-10	74.600	80.000	78.000	85.300
9	10-15-15	66.000	75.300	69.300	80.700
10	10-20-20	67.300	76.000	60.000	79.300
11	15-00-00	95.300	99.300	99.300	100.000
12	15-05-05	78.600	82.600	82.000	90.000
13	15-10-10	74.000	78.600	67.300	79.300
14	15-15-15	69.300	74.000	62.000	79.300
15	15-20-20	58.000	68.700	64.000	77.300
16	20-00-00	98.700	100.000	98.000	100.000
17	20-05-05	88.000	89.300	84.000	90.700
18	20-10-10	76.000	81.300	71.300	82.000
19	20-15-15	64.700	71.300	66.000	74.700
20	20-20-20	62.700	72.000	54.700	70.000
21	25-00-00	97.300	99.300	95.300	100.000
22	25-05-05	80.000	86.700	84.700	90.000
23	25-10-10	71.300	77.300	72.000	83.300
24	25-15-15	70.700	78.700	65.300	78.000
25	25-20-20	58.700	65.300	59.300	72.000
26	30-00-00	94.700	97.300	95.300	100.000
27	30-05-05	84.700	87.300	78.000	87.300
28	30-10-10	72.000	78.700	74.700	82.000
29	30-15-15	60.000	70.000	60.000	74.000
30	30-20-20	54.700	60.700	52.000	66.700
31	35-00-00	87.300	96.700	90.000	100.000
32	35-05-05	70.000	80.000	80.600	96.000
33	35-10-10	72.000	72.700	70.000	82.700
34	35-15-15	62.000	66.700	62.700	73.300
35	35-20-20	52.700	61.300	58.700	73.300
36	40-00-00	89.300	96.000	94.000	100.000
37	40-05-05	78.000	82.700	76.600	85.300
38	40-10-10	66.000	74.000	62.700	79.300
39	40-15-15	58.700	70.700	59.300	76.000
40	40-20-20	50.700	63.300	49.300	72.000
41	45-00-00	88.700	95.300	92.000	100.000
42	45-05-05	72.700	82.700	80.000	90.000
43	45-10-10	56.000	66.000	60.000	80.000
44	45-15-15	55.300	67.300	60.700	78.300
45	45-20-20	50.700	62.000	54.000	66.700
46	50-00-00	82.000	89.300	93.300	100.000
47	50-05-05	75.300	83.300	75.300	90.000
48	50-10-10	60.700	69.300	64.000	79.300
49	50-15-15	53.300	63.300	62.000	73.300
50	50-20-20	46.000	58.700	46.000	66.700

	error-rates	bg-5	bg-5-space	bg-10	bg-10-space
1	05-00-00	100.000	100.000	100.000	100.000
2	05-05-05	87.300	90.700	79.300	91.300
3	05-10-10	74.700	80.000	78.700	91.300
4	05-15-15	65.300	73.300	68.000	84.000
5	05-20-20	71.300	81.300	66.000	81.300
6	10-00-00	100.000	100.000	99.300	100.000
7	10-05-05	77.300	82.000	81.300	88.000
8	10-10-10	74.700	79.300	76.000	81.300
9	10-15-15	70.000	79.300	66.700	78.000
10	10-20-20	60.700	69.300	68.000	83.300
11	15-00-00	99.300	100.000	98.000	100.000
12	15-05-05	85.300	88.000	87.300	90.700
13	15-10-10	73.300	79.300	74.700	86.000
14	15-15-15	60.700	70.700	66.000	78.700
15	15-20-20	62.700	69.300	60.000	68.700
16	20-00-00	98.700	99.300	98.000	100.000
17	20-05-05	82.700	86.000	78.700	84.000
18	20-10-10	71.300	77.300	71.300	84.000
19	20-15-15	66.000	70.700	66.700	78.700
20	20-20-20	63.300	68.000	57.300	71.300
21	25-00-00	98.700	99.300	96.700	100.000
22	25-05-05	79.300	84.700	72.000	83.300
23	25-10-10	70.000	77.300	65.300	79.300
24	25-15-15	66.700	73.300	63.300	76.000
25	25-20-20	60.700	67.300	60.000	74.700
26	30-00-00	96.700	99.300	99.300	100.000
27	30-05-05	78.000	80.000	76.700	89.300
28	30-10-10	64.000	70.000	68.000	81.300
29	30-15-15	57.300	62.700	66.700	79.300
30	30-20-20	56.700	65.300	52.700	65.300
31	35-00-00	92.700	98.000	98.000	100.000
32	35-05-05	73.300	80.000	78.000	89.300
33	35-10-10	70.700	76.700	67.300	80.700
34	35-15-15	64.000	71.300	64.700	78.000
35	35-20-20	52.700	60.700	55.300	68.000
36	40-00-00	91.300	97.300	96.000	100.000
37	40-05-05	78.700	86.700	84.700	91.300
38	40-10-10	68.700	78.700	58.700	75.300
39	40-15-15	52.000	58.700	60.000	75.300
40	40-20-20	53.900	64.700	51.300	72.000
41	45-00-00	87.300	94.000	91.300	99.300
42	45-05-05	78.700	86.000	74.700	88.000
43	45-10-10	60.700	70.700	67.300	80.000
44	45-15-15	59.300	68.700	52.000	68.700
45	45-20-20	48.000	58.000	52.700	70.700
46	50-00-00	87.300	92.700	88.700	98.700
47	50-05-05	75.300	80.700	74.700	87.300
48	50-10-10	53.300	66.700	65.300	78.700
49	50-15-15	52.700	57.300	56.000	71.300
50	50-20-20	46.700	56.700	50.000	70.000

	error-rates	x-wp-5	x-wp-5-space	x-wp-10	x-wp-10-spac
1	05-00-00	100.000	100.000	100.000	100.000
2	05-05-05	80.000	84.000	89.300	96.000
3	05-10-10	74.700	85.300	74.700	86.700
4	05-15-15	65.300	81.300	65.300	84.000
5	05-20-20	62.700	82.700	68.000	81.300
6	10-00-00	98.700	100.000	100.000	100.000
7	10-05-05	85.300	90.700	86.700	94.700
8	10-10-10	73.300	82.700	78.700	89.300
9	10-15-15	66.700	74.700	70.700	84.000
10	10-20-20	58.700	69.300	58.700	88.000
11	15-00-00	100.000	100.000	100.000	100.000
12	15-05-05	84.000	89.300	89.300	94.700
13	15-10-10	73.300	80.000	77.300	92.000
14	15-15-15	64.000	70.700	70.700	86.700
15	15-20-20	58.700	65.300	65.300	85.300
16	20-00-00	94.700	100.000	97.300	100.000
17	20-05-05	84.000	84.000	80.000	86.700
18	20-10-10	69.300	74.700	62.700	81.300
19	20-15-15	58.700	74.700	70.700	85.300
20	20-20-20	56.000	64.000	54.700	74.700
21	25-00-00	96.000	98.700	100.000	100.000
22	25-05-05	82.700	86.700	88.000	96.000
23	25-10-10	66.700	76.000	74.700	89.300
24	25-15-15	57.300	69.300	65.300	85.300
25	25-20-20	57.300	72.000	56.000	78.700
26	30-00-00	97.300	100.000	96.000	100.000
27	30-05-05	78.700	93.300	84.000	90.700
28	30-10-10	76.000	81.300	54.700	77.300
29	30-15-15	58.700	72.000	60.000	82.700
30	30-20-20	53.300	66.700	37.300	73.300
31	35-00-00	89.300	100.000	94.700	100.000
32	35-05-05	85.300	90.700	80.000	89.300
33	35-10-10	68.000	81.300	61.300	80.000
34	35-15-15	66.700	70.700	65.300	74.700
35	35-20-20	49.300	65.300	57.300	72.000
36	40-00-00	96.000	100.000	98.700	100.000
37	40-05-05	78.700	90.700	78.700	90.700
38	40-10-10	65.300	80.000	70.700	86.700
39	40-15-15	60.000	69.300	60.000	80.000
40	40-20-20	60.000	74.700	54.700	77.300
41	45-00-00	96.000	100.000	86.700	100.000
42	45-05-05	78.700	82.700	84.000	96.000
43	45-10-10	54.700	72.000	64.000	81.300
44	45-15-15	52.000	70.700	62.700	92.000
45	45-20-20	42.700	56.000	49.300	70.700
46	50-00-00	86.700	98.700	90.700	100.000
47	50-05-05	69.300	84.000	73.300	90.700
48	50-10-10	66.700	78.700	60.000	86.700
49	50-15-15	56.000	69.300	53.300	73.300
50	50-20-20	48.000	62.700	53.300	69.300

	error-rates	x-bg-5	x-bg-5-space	x-bg-10	x-bg-10-space
1	05-00-00	100.000	100.000	100.000	100.000
2	05-05-05	88.000	90.700	80.000	90.700
3	05-10-10	72.000	84.000	66.700	82.700
4	05-15-15	66.700	80.000	65.300	84.000
5	05-20-20	68.000	80.000	65.300	88.000
6	10-00-00	100.000	100.000	98.700	100.000
7	10-05-05	84.000	88.000	85.300	96.000
8	10-10-10	72.000	81.300	69.300	86.700
9	10-15-15	61.300	69.300	68.000	84.000
10	10-20-20	64.000	76.000	64.000	82.700
11	15-00-00	100.000	100.000	98.700	100.000
12	15-05-05	81.300	85.300	78.700	89.300
13	15-10-10	68.000	78.700	76.000	89.300
14	15-15-15	70.700	82.700	69.300	84.000
15	15-20-20	62.700	69.300	54.700	73.300
16	20-00-00	96.000	100.000	98.700	100.000
17	20-05-05	80.000	86.700	86.700	94.700
18	20-10-10	76.000	80.000	76.000	90.700
19	20-15-15	64.000	77.300	58.700	76.000
20	20-20-20	62.700	69.300	58.700	74.700
21	25-00-00	100.000	100.000	97.300	100.000
22	25-05-05	81.300	88.000	89.300	97.300
23	25-10-10	66.700	73.300	64.000	78.700
24	25-15-15	58.700	73.300	60.000	73.300
25	25-20-20	50.700	61.300	62.700	86.700
26	30-00-00	100.000	100.000	97.300	100.000
27	30-05-05	78.700	84.000	78.700	90.700
28	30-10-10	81.300	88.000	72.000	85.300
29	30-15-15	72.000	73.300	65.300	85.300
30	30-20-20	57.300	65.300	56.000	77.300
31	35-00-00	93.300	100.000	89.300	100.000
32	35-05-05	78.700	88.000	78.700	96.000
33	35-10-10	64.000	70.700	70.700	90.700
34	35-15-15	54.700	68.000	64.000	78.700
35	35-20-20	46.700	60.000	53.300	76.000
36	40-00-00	93.300	100.000	94.700	100.000
37	40-05-05	81.300	85.300	74.700	89.300
38	40-10-10	65.300	73.300	69.300	86.700
39	40-15-15	62.800	70.700	66.700	81.300
40	40-20-20	49.300	56.000	48.000	74.700
41	45-00-00	93.300	98.700	92.000	100.000
42	45-05-05	73.300	82.700	77.300	90.700
43	45-10-10	65.300	73.300	68.000	84.000
44	45-15-15	52.000	58.700	60.000	78.700
45	45-20-20	45.300	64.000	60.000	76.000
46	50-00-00	92.000	97.300	86.700	100.000
47	50-05-05	70.700	85.300	70.700	92.000
48	50-10-10	66.700	78.700	66.700	78.700
49	50-15-15	60.000	73.300	54.700	81.300
50	50-20-20	60.000	68.000	49.300	69.300

Appendix C-3

Phrase Recognition Results
for Cockpit Lexicon

Error rates are listed as substitution, insertion and deletion errors respectively.

The top of the data columns reflect the lexical search space parameters:

wp	-	word pair
bg	-	bi-gram
ss	-	correct phrase found in lexical search space

	error rates	No Grammar	wp	wp w/ ss	bg	bg w/ ss
1	05-00-00	~ 35.0	93.200	94.600	95.500	96.300
2	05-05-05		89.500	90.300	91.900	93.200
3	05-10-10		87.800	89.200	86.500	87.800
4	05-15-15		83.800	85.100	81.800	83.800
5	05-20-20		82.400	84.500	83.800	85.100
6	10-00-00		89.500	91.000	88.800	89.600
7	10-05-05		75.100	76.900	76.300	78.400
8	10-10-10		74.300	74.300	75.700	75.700
9	10-15-15		68.900	71.000	68.200	70.300
10	10-20-20		58.800	60.800	58.800	60.100
11	15-00-00		85.100	85.800	84.300	85.800
12	15-05-05		58.900	59.700	58.800	59.500
13	15-10-10		56.700	58.100	58.700	59.500
14	15-15-15		52.000	52.700	52.000	52.700
15	15-20-20		48.600	49.300	56.700	58.800
16	20-00-00		79.800	82.800	78.400	79.800
17	20-05-05		44.000	44.000	50.700	51.300
18	20-10-10		41.200	42.500	51.300	51.300
19	20-15-15		42.600	43.200	42.600	43.200
20	20-20-20		38.500	39.200	44.600	45.900
21	25-00-00		73.900	76.100	66.400	70.100
22	25-05-05		40.300	41.000	44.600	45.300
23	25-10-10		37.800	37.800	38.500	38.500
24	25-15-15		29.000	29.700	29.000	29.700
25	25-20-20		31.700	33.100	30.400	31.000
26	30-00-00		65.700	68.700	59.700	62.700
27	30-05-05		35.800	36.500	29.000	29.700
28	30-10-10		23.000	23.600	28.400	29.700
29	30-15-15		24.300	25.700	22.300	23.600
30	30-20-20		19.600	19.600	21.600	21.600
31	35-00-00		56.700	60.400	58.100	60.100
32	35-05-05		30.400	30.400	25.700	26.300
33	35-10-10		20.900	20.900	20.900	20.900
34	35-15-15		19.600	20.300	13.500	13.500
35	35-20-20		14.200	15.500	20.900	22.300
36	40-00-00		57.500	61.900	56.700	58.800
37	40-05-05		20.900	21.600	18.200	18.900
38	40-10-10		17.600	17.600	13.500	14.200
39	40-15-15		16.900	17.600	14.200	15.500
40	40-20-20		9.400	10.100	12.800	13.500
41	45-00-00		46.200	48.500	51.300	54.700
42	45-05-05		20.200	21.600	16.200	17.600
43	45-10-10		12.800	13.500	12.200	12.800
44	45-15-15		16.900	17.600	13.500	14.100
45	45-20-20		10.100	10.100	7.400	8.100
46	50-00-00		45.900	48.000	45.300	49.300
47	50-05-05		12.100	12.800	14.200	14.900
48	50-10-10		8.700	9.400	11.500	12.200
49	50-15-15		8.800	8.800	8.100	8.100
50	50-20-20		5.400	6.100	5.400	6.800

Appendix D-1

Digit Lexicon

```
(((z iy r ow) zero)
 ((w ah n) one)
 ((t uw) two)
 ((th r iy) three)
 ((f ow r) four)
 ((f ah ih v) five)
 ((s ih k s) six)
 ((s eh v ax n) seven)
 ((eh ih t) eight)
 ((n ah ih n) nine))
```

Appendix D-2

Cockpit Lexicon

((a eh ih)
(about ax b ah uw t)
(abreast ax b r eh s t)
(accelerate eh k s eh l er eh ih t)
(accept eh k s eh p t)
(accepted ae k s eh p t ix d)
(access ae k s eh s)
(accomplish ax k aa m p l ix sh)
(acknowledge ae k n aa l ix jh)
(action ae k sh ax n)
(activate ae k t ix v eh ih t)
(active ae k t ix v)
(activity ax k t ih v ix t iy)
(additional ax d ih sh ax n eh l)
(advise ax d v ah ih z)
(against ax g eh n s t)
(ahead ax hh eh d)
(aid eh ih d)
(aim eh ih m)
(aims eh ih m z)
(air eh r)
(airborne eh r b ow r n)
(aircraft eh r k r ax f t)
(airplane eh r p l eh ih n)
(airplanes eh r p l eh ih n z)
(all ao l)
(allied ae l ah ih d)
(alter ao l t er)
(alternate ao l t er n ax t)
(alternates ao l t er n ax t s)
(alternative ao l t er n ix t ix v)
(altitude ae l t ix t uw d)
(am ae m)
(amraam ae m r ae m)
(amraams ae m r ae m z)
(an ae n)
(analysis ax n ae l ix s ix s)
(analyze ae n ax l ah ih z)
(and ae n d)
(any eh n iy)
(a-p-x eh ih p iy eh k s)
(are aa r)
(area eh ih r iy ax)

(arm aa r m)
(armament aa r m ax m ax n t)
(armed aa r m d)
(arms aa r m z)
(around ax r ah uw n d)
(as ae z)
(aspect ae s p eh k t)
(assess ax s eh s)
(assessment ax s eh s m ax n t)
(assign ax s ah ih n)
(assignment ax s ah ih n m ax n t)
(assignments ax s ah ih n m ax n t s)
(at ae t)
(attach ax t ae ch)
(attack ax t ae k)
(attackers ax t ae k er z)
(attempt ax t eh m p t)
(attempted ax t eh m p t ix d)
(attempts ax t eh m p t s)
(auto ao t ow)
(automatic ao t ax m ae t ix k)
(available ax v eh ih l ax b eh l)
(avionics eh ih v iy aa n ix k s)
(avoid ax v ow iy d)
(avoidance ax v ow iy d ax n s)
(avoiding ax v ow iy d ix ng)
(back b ae k)
(band b ae n d)
(bandit b ae n d ix t)
(bandits b ae n d ix t s)
(base b eh ih s)
(bases b eh ih s ix z)
(battle b ae t eh l)
(b-d b iy d iy)
(b-d-a b iy d iy eh ih)
(beam b iy m)
(bearing b eh ih r ix ng)
(begin b ax g ih n)
(best b eh s t)
(better b eh t er)
(big b ih g)
(bit b ih t)
(blind b l ah ih n d)

(blowup b l ow ah p)
(bogey b ow g iy)
(bogeys b ow g iy z)
(bomber b aa m er)
(bombers b aa m er z)
(bore b ow r)
(both b ow th)
(box b aa k s)
(break b r eh ih k)
(breakout b r eh ih k ah uw t)
(bring b r ih ng)
(bug b ah g)
(burst b er s t)
(buzzers b ah z er z)
(b-v-r b iy v iy aa r)
(by b ah ih)
(c s iy)
(calculate k ae l k y eh l eh ih t)
(call k ao l)
(can k ae n)
(chaff ch ae f)
(change ch eh ih n jh)
(changes ch eh ih n jh ix z)
(check ch eh k)
(chief ch iy f)
(choose ch uw z)
(chopper ch aa p er)
(clear k l iy r)
(cleared k l iy r d)
(climb k l ah ih m)
(close k l ow z)
(closer k l ow s er)
(closest k l ow s ix s t)
(closure k l ow zh er)
(collision k eh l ih zh ax n)
(combat k aa m b ae t)
(come k ah m)
(command k ax m ae n d)
(commence k ax m eh n s)
(commit k ax m ih t)
(complete k ax m p l iy t)
(configure k ax n f ih g y er)
(confirm k ax n f er m)

(confirmed k ax n f er m d)
(consent k ax n s eh n t)
(continue k ax n t ih n uw)
(control k ax n t r ow l)
(conversion k ax n v er zh ax n)
(count k ah uw n t)
(counter k ah uw n t er)
(countermeasure k ah uw n t er m eh zh y er)
(countermeasures k ah uw n t er m eh zh y er z)
(course k ow r s)
(criteria k r ah ih t iy r iy ax)
(cross k r ao s)
(crossing k r ao s ix ng)
(cruise k r uw z)
(cue k uw)
(damage d ae m ix jh)
(danger d eh ih n jh er)
(data d eh ih t ax)
(defeat d ix f iy t)
(defend d ix f eh n d)
(defense d ax f eh n s)
(defensive d ix f eh n s ix v)
(define d ix f ah ih n)
(deploy d ix p l ow iy)
(describe d ix s k r ah ih b)
(designate d eh z ix g n eh ih t)
(designation d ix s ix g n eh ih sh ax n)
(detail d iy t eh ih l)
(details d iy t eh ih l z)
(detection d ix t eh k sh ax n)
(direct d er eh k t)
(direction d er eh k sh ax n)
(dispense d ix s p eh n s)
(display d ix s p l eh ih)
(divert d ax v er t)
(divert d ah ih v er t)
(do d uw)
(dogfight d ao g f ah ih t)
(doing d uw ix ng)
(dope d ow p)
(drop d r aa p)
(east iy s t)
(e-c-m iy s iy eh m)

(egress iy g r eh s)
(eighteen eh ih t iy n)
(element eh l ax m ax n t)
(them th eh m)
(employ ax m p l ow iy)
(encounter ix ng k ah uw n t er)
(endurance ix n d uw r ax n s)
(enemy eh n eh m iy)
(engage ix ng g eh ih jh)
(engaged ix ng g eh ih jh d)
(engagement ix ng g eh ih jh m ax n t)
(enlarge ax n l aa r jh)
(enter eh n t er)
(envelope eh n v eh l ow p)
(equipment ax k w ih p m ax n t)
(escort eh s k ow r t)
(evaluate ix v ae l uw eh ih t)
(evasion ax v eh ih zh ax n)
(evasive ix v eh ih s ix v)
(everybody eh v r iy b ah d iy)
(everything eh v r iy th ix ng)
(execute eh k s eh k uw t)
(expand ax k s p ae n d)
(expend ax k s p eh n d)
(expendables ax k s p eh n d ax b eh l z)
(express ax k s p r eh s)
(eye ah ih)
(fast f ae s t)
(feet f iy t)
(fence f eh n s)
(fifty f ih f t iy)
(fight f ah ih t)
(fighter f ah ih t er)
(fighters f ah ih t er z)
(file f ah ih l)
(fire f ah ih r)
(five f ah ih v)
(flare f l eh r)
(flares f l eh r z)
(flight f l ah ih t)
(fly f l ah ih)
(foe f ow)
(follow f aa l ow)

(following f aa l ow ix ng)
(for f ao r)
(forget f ow r g eh t)
(formation f ow r m eh ih sh ax n)
(forty f ow r t iy)
(four f ow r)
(friend f r eh n d)
(friendlies f r eh n d l iy z)
(friendly f r eh n d l iy)
(front f r ah n t)
(fuel f uw eh l)
(full f uh l)
(future f uw ch er)
(gas g ae s)
(g-c-i jh iy s iy ah ih)
(gear g iy r)
(geometry jh iy aa m ax t r iy)
(get g eh t)
(give-me g ih m iy)
(give g ih v)
(go g ow)
(gods g aa d z)
(going g ow ih ng)
(good g uh d)
(got g aa t)
(granted g r ae n t ix d)
(green g r iy n)
(ground g r ah uw n d)
(gun g ah n)
(guns g ah n z)
(guys g ah ih z)
(hahn hh aa n)
(have hh ae v)
(he hh iy)
(heading hh eh d ix ng)
(heat hh iy t)
(heater hh iy t er)
(helicopter hh eh l ax k aa p t er)
(helo hh iy l ow)
(help hh eh l p)
(hide hh ah ih d)
(high hh ah ih)
(highest hh ah ih ax s t)

(highlight hh ah ih l ah ih t)
(him hh ih m)
(history hh ih s t er iy)
(hold hh ow l d)
(home hh ow m)
(homeplate hh ow m p l eh ih t)
(hook hh uh k)
(horizontal hh ow r ix z aa n t eh l)
(hostile hh aa s t ah ih l)
(hostiles hh aa s t ah ih l z)
(hot hh aa t)
(how hh ah uw)
(hows hh ah uw z)
(had hh ae d)
(h-s-d eh ih ch s d iy)
(h-t-e eh ih ch t iy iy)
(hundred hh ah n d r ax d)
(i ah ih)
(i-d ah ih d iy)
(identification ah ih d ax n t ix f ix k eh ih sh ax n)
(identify ah ih d eh n t ix f ah ih)
(i-m ah ih eh m)
(i-ve ah ih v)
(impact ix m p ae k t)
(implement ih m p l ax m ax n t)
(in ih n)
(info ih n f ow)
(inform ix n f ow r m)
(information ix n f er m eh ih sh ix n)
(infra ih n f r ax)
(ingress ih ng g r ax s)
(initiate ix n ih sh iy eh ih t)
(inroute ih n r uw t)
(instructions ix n s t r ah k sh ax n z)
(intercept ih n t er s eh p t)
(interrogate ix n t eh ih r ax g eh ih t)
(into ih n t uw)
(i-r ah ih aa r)
(irst er s t)
(i-r-s-t ah ih aa r eh s t iy)
(is ih z)
(it ih t)
(I-have ah ih v)

(j jh eh ih)
(jam jh ae m)
(jammer jh ae m er)
(jammers jh ae m er z)
(jamming jh ae m ix ng)
(jazz jh ae z)
(join jh ow iy n)
(keep k iy p)
(kill k ih l)
(killer k ih l er)
(kind k ah ih n d)
(know n ow)
(launch l ao n sh)
(lead l iy d)
(leader l iy d er)
(lean l iy n)
(left l eh f t)
(let l eh t)
(lethal l iy th eh l)
(lets l eh t s)
(level l eh v eh l)
(like l ah ih k)
(lima l iy m ax)
(lima l ah ih m ax)
(line l ah ih n)
(link l ih ng k)
(load l ow d)
(location l ow k eh ih sh ax n)
(lock l aa k)
(locked l aa k t)
(long l ao ng)
(look l uh k)
(low l ow)
(l-r-s eh l aa r eh s)
(mach m aa k)
(magnum m ae g n ax m)
(main m eh ih n)
(man m ae n)
(maneuver m ax n uw v er)
(manual m ae n uw eh l)
(map m ae p)
(master m ae s t er)
(max m ae k s)

(me m iy)
(medium m iy d iy ax m)
(mein m ah ih n)
(members m eh m b er z)
(message m eh s ix jh)
(mig m ih g)
(miles m ah ih l z)
(missile m ih s eh l)
(missiles m ih s eh l z)
(mission m ih sh ax n)
(mode m ow d)
(monitor m aa n ix t er)
(more m ow r)
(mothers m ah d hh er z)
(move m uw v)
(m-r-m eh m aa r eh m)
(mud m ah d)
(multiple m ah l t ix p eh l)
(my m ah ih)
(narrow n eh r ow)
(nav n ae v)
(navigation n ae v ix g eh ih sh ax n)
(nearest n iy r ax s t)
(negative n eh g ix t ix v)
(net n eh t)
(new n uw)
(newest n uw ax s t)
(nine n ah ih n)
(ninety n ah ih n t iy)
(non n aa n)
(north n ow r th)
(nose n ow z)
(notify n ow t ax f ah ih)
(now n ah uw)
(number n ah m b er)
(numbers n ah m b er z)
(oclock ax k l aa k)
(of ax v)
(off ao f)
(offense ao f eh n s)
(offensive ao f eh n s ix v)
(offset ao f s eh t)
(ok ow k eh ih)

(on aa n)
(one w ah n)
(ones w ah n z)
(only ow n l iy)
(operational aa p er eh ih sh ax n eh l)
(ops aa p s)
(optimal aa p t ix m eh l)
(optimize aa p t ix m ah ih z)
(optimum aa p t ix m ax m)
(option aa p sh ax n)
(options aa p sh ax n z)
(or ow r)
(ordnance ow r d n ax n s)
(other ah d hh er)
(our aa r)
(out ah uw t)
(overall ow v er ao l)
(overview ow v er v uw)
(package p ae k ix jh)
(parameters p er ae m ax t er z)
(parrot p eh r ax t)
(pass p ae s)
(passive p ae s ix v)
(path p ae th)
(perform p er f ow r m)
(performance p er f ow r m ax n s)
(picture p ih k ch er)
(pigeons p ih jh ax n z)
(pin p ih n)
(pincer p ih n s er)
(pinch p ih n sh)
(p-k p iy k eh ih)
(plan p l ae n)
(planes p l eh ih n z)
(plate p l eh ih t)
(platform p l ae t f ow r m)
(plot p l aa t)
(pod p aa d)
(pods p aa d z)
(point p ow iy n t)
(posit p aa s ix t)
(position p ax s ih sh ax n)
(prepare p r iy p eh r)

(present p r eh s ax n t)
(presentation p r eh s ax n t eh ih sh ax n)
(press p r eh s)
(primary p r ah ih m er iy)
(primary p r ah ih m eh r iy)
(priorities p r ah ih ow r ix t iy z)
(prioritize p r ah ih ow r ix t ah ih z)
(prioritized p r ah ih ow r ix t ah ih z d)
(priority p r ah ih ao r ix t iy)
(probability p r aa b ax b ih l ix t iy)
(proceed p r ax s iy d)
(profile p r ow f ah ih l)
(profiles p r ow f ah ih l z)
(program p r ow g r ae m)
(project p r aa jh eh k t)
(put p uh t)
(quick k w ih k)
(radar r eh ih d aa r)
(radio r eh ih d iy ow)
(radius r eh ih d iy ax s)
(raid r eh ih d)
(rain r eh ih n)
(ram r ae m)
(ramstein r ae m s t ah ih n)
(range r eh ih n jh)
(raw r ao)
(reading r iy d ix ng)
(ready r eh d iy)
(reassign r iy ix s ah ih n)
(reassignment r iy ix s ah ih n m ax n t)
(recalculate r iy k ae k uw l eh ih t)
(reconfigure r ix k ix n f ih g y er)
(recover r ix k ah v er)
(recovery r ix k ah v er iy)
(red r eh d)
(rejoin r iy jh ow iy n)
(relative r eh l ix t ix v)
(relay r eh l eh ih)
(relay r iy l eh ih)
(remode r ix m ow d)
(rendezvous r aa n d ix v uw)
(repeat r ix p iy t)
(report r ix p ow r t)

(request r ix k w eh s t)
(request r iy k w eh s t)
(reroute r iy r uw t)
(reroute r iy r ah uw t)
(resort r ix z ow r t)
(rest r eh s t)
(retarget r iy t aa r g ax t)
(retargeting r ix t aa r g ih t ix ng)
(return r ix t er n)
(return r iy t er n)
(rhein r eh ih n)
(right r ah ih t)
(ring r ih ng)
(route r ah uw t)
(routes r ah uw t s)
(routing r ah uw t ix ng)
(r-t-b aa r t iy b iy)
(r-v aa r v iy)
(r-w-r aa r d ah b eh l uw aa r)
(s-a eh s eh ih)
(saber s eh ih b er)
(safe s eh ih f)
(safest s eh ih f ax s t)
(salvo s ae l v ow)
(sam s ae m)
(sample s ae m p eh l)
(say s eh ih)
(scale s k eh ih l)
(scan s k ae n)
(scanners s k ae n er z)
(schedule s k eh d y eh l)
(scope s k ow p)
(screen s k r iy n)
(search s er ch)
(seater s iy t er)
(seconds s eh k ax n d z)
(see s iy)
(select s eh l eh k t)
(selected s eh l eh k t ix d)
(selection s eh l eh k sh ax n)
(self s eh l f)
(send s eh n d)
(sensor s eh n s er)

(separate s eh p ax r ax t)
(set s eh t)
(setting s eh t ix ng)
(seven s eh v ax n)
(share sh eh r)
(shirk sh er k)
(shoot sh uw t)
(short sh ow r t)
(shot sh aa t)
(show sh ow)
(sided s ah ih d ix d)
(sidewinder s ah ih d w ah ih n d er)
(signal s ih g n eh l)
(signature s ih g n ax ch er)
(simultaneous s ih m eh l t ae n iy ax s)
(simultaneously s ah ih m eh l t eh ih n iy ax s l iy)
(single s ih ng g eh l)
(site s ah ih t)
(situation s ih ch uw eh ih sh eh n)
(six s ih k s)
(sixty s ih k s t iy)
(snakes s n eh ih k s)
(snap s n ae p)
(sort s ow r t)
(sorted s ow r t ix d)
(sparkle s p aa r k eh l)
(sparrow s p eh r ow)
(speak s p iy k)
(specific s p ax s ih f ix k)
(specify s p eh s ax f ah ih)
(speed s p iy d)
(split s p l ih t)
(spotlight s p aa t l ah ih t)
(s-r-m eh s aa r eh m)
(standby s t ae n d b ah ih)
(start s t aa r t)
(state s t eh ih t)
(stats s t ae t s)
(status s t eh ih t ax s)
(steer s t iy r)
(steering s t iy r ix ng)
(store s t ow r)
(stores s t ow r z)

(strike s t r ah ih k)
(striker s t r ah ih k er)
(strikers s t r ah ih k er z)
(stripped s t r ih p t)
(sub s ah b)
(suckers s ah k er z)
(suggest s ax g jh eh s t)
(suitable s uw t ax b eh l)
(swap s w aa p)
(sweep s w iy p)
(swing s w ih ng)
(switch s w ih ch)
(system s ih s t ax m)
(systems s ih s t ax m z)
(tactic t ae k t ix k)
(tactical t ae k t ix k eh l)
(tactics t ae k t ix k s)
(take t eh ih k)
(tankers t ae ng k er z)
(target t aa r g ix t)
(targeted t aa r g ix t ix d)
(targeting t aa r g ix t ix ng)
(targets t aa r g ix t s)
(tell t eh l)
(ten t eh n)
(terrain t er eh ih n)
(test t eh s t)
(t-f t iy eh f)
(t-f-r t iy eh f aa r)
(t-f-t-a t iy eh f t iy eh ih)
(that dh ae t)
(the dh ax)
(their dh eh r)
(them dh eh m)
(there dh eh r)
(they dh eh ih)
(thirty th er t iy)
(those dh ow z)
(threat th r eh t)
(threats th r eh t s)
(three th r iy)
(through th r uw)
(time t ah ih m)

(to t uw)
(t-o-t t ow t)
(track t r ae k)
(tracking t r ae k ix ng)
(trail t r eh ih l)
(trailer t r eh ih l er)
(transfer t r ae n s f er)
(transmit t r ax n z m ih t)
(trouble t r ah b eh l)
(turn t er n)
(twelve t w eh l v)
(twenty t w eh n t iy)
(two t uw)
(twos t w aa z)
(t-w-s t iy d ah b eh l uw eh s)
(type t ah ih p)
(unknown ax n n ow n)
(up ah p)
(update ah p d eh ih t)
(updated ah p d eh ih t ix d)
(updated ah p d eh ih dx ix d)
(updating ah p d eh ih t ix ng)
(us ax s)
(vector v eh k t er)
(vectors v eh k t er z)
(verbalize v er b ax l ah ih z)
(vid v ih d)
(view v uw)
(visual v ih zh uw eh l)
(v-sub-c v iy s ah b s iy)
(want-to w aa n ax)
(want w aa n t)
(way w eh ih)
(we w iy)
(weapon w eh p ax n)
(weapons w eh p ax n z)
(well w eh l)
(were w er)
(what w ax t)
(whats w ax t s)
(when w eh n)
(where w eh r)
(wheres w eh r z)

(which w ih ch)
(while w ah ih l)
(wide w ah ih d)
(wilco w ih l k ow)
(will w ih l)
(window w ih n d ow)
(wing w ih ng)
(wingman w ih ng g m ax n)
(wingmen w ih ng g m eh n)
(wingmans w ih ng g m ax n z)
(with w ih th)
(working w er k ix ng)
(worst w er s t)
(yes y eh s)
(you uw)
(your y er)
(you-ve y uw v)
(zap z ae p)
(zero z iy r ow)
(zone z ow n)
(zoom z uw m))