Natural-Sounding Speech Synthesis Using Variable-Length Units

by

Jon Rong-Wei Yi

S.B., Massachusetts Institute of Technology, 1997

Submitted to the Department of Electrical Engineering
and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1998

Author . . . .
Department of Electrical Engineering
and Computer Science
May 21, 1998

Certified by . . . .
James R. Glass
Principal Research Scientist
Thesis Supervisor

Accepted by . . . .
Arthur C. Smith
Chairman, Departmental Committee on Graduate Students

Natural-Sounding Speech Synthesis Using Variable-Length Units

by

Jon Rong-Wei Yi

Submitted to the Department of Electrical Engineering
and Computer Science
on May 21, 1998 in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

The goal of this work was to develop a speech synthesis system which concatenates variable-length units to create natural sounding speech. Our initial work in this area showed that by careful design of system responses to ensure consistent intonation contours, natural-sounding speech synthesis was achievable with word- and phrase-level concatenation. In order to extend the flexibility of this framework, subsequent work focused on the problem of generating novel words from a pre-recorded corpus of sub-word units. The design of the sub-word units was motivated by perceptual experiments that investigated where speech could be spliced with minimal distortion and what contextual constraints were necessary to maintain in order to produce natural sounding speech. This sub-word corpus is then searched at synthesis time with a Viterbi search which selects a sequence of units based on how well they individually match the input specification and on how well they sound as an ensemble. This concatenative speech synthesis system, ENVOICE, has been used in a conversational information retrieval system in two application domains to convert meaning representations into speech waveforms.

Thesis Supervisor: James R. Glass
Title: Principal Research Scientist

Acknowledgments

First, I would like to thank Jim for all the time, energy, and efforts he has directed towards this work, a capstone experience representing the four years I have spent at SLS so far. After serving as a mentor for several of my UROP projects and my Bachelor's thesis, he once again guided me along the road of research for my M.Eng. thesis. With his unrelenting, yet gentle encouragement, he made certain I never veered off the track - for too long. I am indebted to his ability and willingness to pre-empt his many other tasks time after time to work with me, be it to help with research, or writing.

I would also like to thank Helen and Stephanie for the many enlightening discussions about syllable theory and syllabification algorithms. I am grateful to Helen, T.J., and Chao for taking the time to read through this thesis, and for the many improvements brought about by their suggestions.

I would like to offer my appreciation to the people who donated their voices to this thesis. In the initial work with WHEELS, Vicky provided the seed corpus. Later on, Tom Lee at BellSouth recorded the final corpus. For the latter work, Michelle recorded the sub-word corpus, and PEGASUS system responses and carrier phrases.

I would like to thank Christine, Helen, Stephanie, Joe, and Jim for assisting in the integration of this work into GALAXY. Christine designed the speech synthesizer server protocol and client/server interface. The WHEELS back-end designed by Helen was the first consumer of the phrase-level concatenative synthesizer. Stephanie and Joe designed the PEGASUS back-end which was the first consumer of the sub-word synthesis framework. In both of these systems, Jim assisted with the corpus design, recording, and transcription.

I thank Victor and all of the members of SLS for the stimulating working atmosphere they have provided, and for the comments and suggestions they offered in forming a slide presentation for this work.

Finally, I am forever grateful to my family who has supported me in my five years at MIT. Since coming to college, the time I have spent with them has become less, but the love has never diminished. I thank my mother and father for their nurturing and loving care. I thank my two younger brothers for always reminding me of our happy childhood together. I thank all my relatives for the close-knit relationships we have developed.

This research was supported by DARPA under Contract N66001-96-C-8256, monitored through Naval Command, Control and Ocean Surveillance Center, and by a research contract from BellSouth Intelliventures.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In a perfect world, a speech synthesizer should be able to synthesize any arbitrary word sequence with perfect intelligibility and naturalness. As shown in Figure 1-1, current synthesizers have tended to strive for flexibility of vocabulary and sentences at the expense of naturalness (i.e., arbitrary words can be synthesized, but do not sound very natural.) This applies to articulatory, terminal-analog, and concatenative methods of speech synthesis [2, 4, 8, 16, 17, 22, 28].

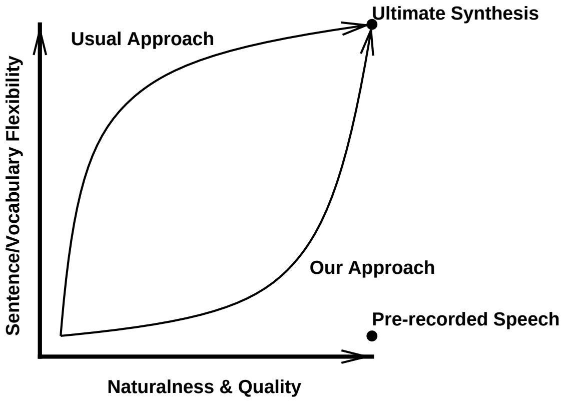


Figure 1-1: Synthesis development curve in this thesis work.

An alternative strategy is one which seeks to maintain naturalness by operating in a constrained domain. There are potentially many applications where this mode of operation is perfectly suitable. In speech understanding systems for example, the domain of operation is often quite limited, and is known ahead of time.

An extreme example of maintaining naturalness is the use of pre-recorded speech. A step beyond this is word- or phrase-level concatenation of speech segments from pre-recorded utterances. As we wish to increase word flexibility, we turn to concatenating together ever smaller-sized units. A decrease in unit length must be accompanied by an increase of context. The selection of which units to use in concatenation is a process guided by contextual information to preserve co-articulatory and prosodic constraints. Past works by others have examined how unit selection algorithms can be formulated, and what constraints must be maintained [2, 4, 17, 28].

In this thesis, we develop a framework for natural sounding speech synthesis using variable length units. Our developmental philosophy that we have adhered to throughout the thesis research places naturalness as the paramount goal. We achieve this by performing Meaning-to-Speech (MTS) synthesis in constrained application domains.

In our preliminary work involving phrase-level concatenation, the vocabulary size is relatively small, but naturalness is quite high. After the sub-word concatenation system is developed, new words can be manufactured from units in the sub-word unit database. This gives us the ability to create new vocabulary by naturally concatenating together speech segments which may happen to be non-contiguous. Our research follows the bottom curve in Figure 1-1 of where we view naturalness as the highest priority, while steadily increasing sentence and vocabulary flexibility. As the pursuit of naturalness dominates, human listening provides the best feedback. However, we do attempt to take objective measurements as well.

1.1 Background

Concatenative speech synthesis algorithms are currently popular, in that they are relatively simple compared to articulatory or terminal-analog methods [8, 22] and can produce natural-sounding speech. Recently, time-domain concatenative speech synthesis algorithms have become prominent, because of their low computational requirements compared to frequency-based methods; in their simplest form, they only involve playing back speech segments in a concatenated manner, which requires no signal processing, nor floating-point operations.

Past work in unit selection, the process of selecting an optimal sequence of units from a speech database, has focused on searching unit graphs with a distance metric consisting of two costs: a target and a concatenation cost [2, 4, 17, 28]. Target costs can incorporate information about phonological environment, spectral measures, and prosody measures. Concatenation costs can incorporate information about spectral continuity and prosody continuity. It can also contain trigram context in the form of triphones [16]. However, most of these works have operated over a generic speech corpus not specifically designed for concatenative speech synthesis. In this thesis, we attempt to define an appropriate set of units, and design a corpus which can be used for synthesis of isolated words.

Because concatenative speech synthesis algorithms can merely abut together non-contiguous speech segments, the resulting prosody, which encompasses the three dimensions of pitch, duration, and energy, may not necessarily sound natural. This can be corrected after the fact by prosodic modification algorithms. An example of an algorithm which happens to operate in the time domain is the Time-Domain Pitch-Synchronous Overlap-and-Add algorithm (TD-PSOLA) [5, 14].

Past research in prosody suggests that intonation phrases exist in two types: those with and without boundary tones to the right side [26]. Therefore, it may be possible to extract and splice intonation phrases as long as knowledge of the presence of boundary tones is preserved. These intonation building blocks can also be thought of *High* and *Low* phrases [29]. In this thesis, we do not actively address the issue of prosodic modification. Instead we use knowledge of the constrained meaning representation to help design natural-sounding phrase-level concatenation augmented with proper names synthesized from sub-word units.

1.2 Outline

In Chapter 2, we will describe preliminary work which focused on the design of a word- and phrase-level synthesizer called ENVOICE. It converts an internal meaning representation into a speech waveform for system responses and spoken information in GALAXY, a conversational speech system [13]. This is a two-step process in which a recursive generation system, GENESIS [11], converts a meaning representation into text using a pre-defined set of vocabulary and message templates. ENVOICE utilizes GENESIS with a set of vocabulary and message templates annotated with speech segment descriptions to generate a complete speech segment description from a meaning representation. Then, ENVOICE concatenates the specified speech segments for the final speech waveform output.

As part of system responses to human queries, pre-fabricated carrier phrases are filled in with pre-recorded words. The intonation contours of the carrier phrases are planned such that individual words, or phrases can be spliced in with little perceptual aberration. This represents the highest quality synthesized speech possible without recording every combination of carrier phrases and vocabulary items. Also, this shows that maintaining prosody constraints at the word level is more important than fulfilling co-articulatory constraints across words. However, this requires the enumeration and recording in advance of all possible response phrases and vocabulary words. Furthermore, if the vocabulary needed to be expanded, new additional words would be need to be recorded. This is somewhat akin to the new word problem in speech recognition.

In the bulk of this thesis work, the synthesis of these "new words" is considered. This task of word synthesis is achieved using sub-word units. The sub-word concatenation system will focus on fulfilling co-articulatory constraints at the sub-word level which will be seen to be more important than fulfilling prosody constraints. In Chapter 3, we will describe perceptual experiments in which we determine where speech can be spliced, or concatenated with minimal perceptual distortions. These experiments help us to gain knowledge about co-articulatory constraints. We separate the constraints into two types, a *unit criterion* and a *transition criterion.* The former describes how individual sub-word units are optimal when compared to a synthesis request. The latter describes how an entire sequence of sub-word units is optimal as an ensemble. The unit and transition criterion can also include optional prosodic measures to score the prosody match at unit boundaries and across unit boundaries, respectively.

Next in Chapter 4, we create an inventory of units based on the contextual constraints learned from the perceptual experiments. The design of an inventory primarily depends on the phonological taxonomy of the language at hand [1]. Also, it should take into account various factors ranging from syllable structure, to place of articulation, to lexical stress. In addition, the units must capture as much of the co-articulation effects as possible in the interior portions, thereby driving optimal splice points to the unit boundaries. The units will contain contextual information in their labels which are used as indexing dimensions in the unit and transition cost functions. This context would be used in calculating the cost of concatenating two arbitrary units.

Once the inventory of sub-word units is enumerated, a prompt selection algorithm will be introduced to automatically select a set of words for recording that compactly cover all the units. This will form the sub-word corpus. Ultimately, if compactness can be traded off, when the units are collected, the two following statements hold true: 1) insuring prosodic variety at recording time can help to keep unit costs due to prosodic mismatch, and 2) acquiring units in many different permutations can help to keep transition costs down.

Once the inventory of units is enumerated, and the sub-word corpus recorded, the final component of the sub-word concatenation system will be delineated in Chapter 5, the unit selection algorithm. It is the role of the search algorithm to select a sequence of units that minimizes a cost function of the designer's choice. We shall formulate our cost function - as past work by others have - in terms of two decoupled costs: a unit and transition cost [2, 17, 4, 28]. While this work intends to create a unit selection algorithm which can search over a general speech corpus, it also intends to contribute something novel: the design of a corpus especially intended for sub-word concatenation. This does not represent the blind application of a search algorithm to any speech corpus, but involves high-level design from the speech researcher.

The final sub-word concatenation system will draw units from a unit database whose constituent words are automatically selected by a prompt selection algorithm which compactly covers the units enumerated in the design process. This sub-word corpus is compiled into a database for fast searching at run-time. This sub-word system outputs speech waveform given a pronunciation as input. The tuning of the sub-word concatenation system will be performed by the researcher in a synthesis-analysis cycle. This development process is assisted by three tools for the creation, viewing, and evaluation of synthesis waveforms.

Chapter 2

Phrase-level concatenation

2.1 Introduction

This chapter describes preliminary work that led up to the development of a sub-word concatenation framework. The synthesis process involves the concatenation of word- and phrase-level units with no signal processing. These units are carefully prepared by recording them in the precise prosodic environment in which they will be used. As we shall see, this type of unit design and unit concatenation achieves a high level of naturalness.

This chapter will introduce the concept of a *carrier phrase*, a template which constrains the prosody into a consistent pattern. Carrier phrases are used as vehicles for recording words and phrases and as the basis for synthesis. Because we record vocabulary items only once, vocabulary flexibility is zero by definition, but we can instantiate the carrier phrases with different vocabulary items in a constrained order, so sentence flexibility is moderate.

By choosing to work in a constrained application domain with a small vocabulary, it is practical to record every word in every prosodic environment realizable. This is a trade-off between large-scale recording and high naturalness. From the synthesizer development curve in Figure 1-1, this corresponds to a partial traversal of the bottom development curve where overall flexibility is adequate (zero vocabulary flexibility and high sentence flexibility), yet naturalness is high.

In this chapter, we will present the usage of phrase-level concatenation for response generation from a meaning representation in WHEELS [21], a conversational information retrieval system. The domain is automobile classified advertisements. The types of responses are finite and each type has an associated carrier phrase. The vocabulary size is on the order of 1,200 words.

2.2 Response generation

In conversational language systems, responses can be generated from an internal, high-level meaning representation or *semantic frame*. These frames store all the information pertinent to the current query, or database record as key-value pairs; these pairs can be possibly recursive, and the values can possibly be another frame.

A meaning representation can be converted to a surface representation, such as text, by means of a generation system. In this work, we make use of GENESIS [11], a system that recursively builds up a sentence given a meaning representation and a message template. This message template then draws on elements from the frame and a pre-defined vocabulary to generate the sentence. Although the key-value pairs may be stored in English within the meaning representation, vocabulary lookups are performed at generation time. This lends to GENESIS multi-lingual generation capabilities. We will make use of this ability to insert ENVOICE-specific annotations.

In Figure 2-1 we see a flowchart for the process of phrase-level concatenation. A meaning representation input is converted to speech waveform output by the component block containing GENESIS and ENVOICE. Internally, this is a two-step process:

GENESIS converts the meaning representation using the pre-defined vocabulary and message templates into an annotated description of waveform segments. ENVOICE then takes this description of word- and phrase-level segments and performs concatenation using waveform segments from the database. The final speech waveform output undergoes no signal processing in this configuration.

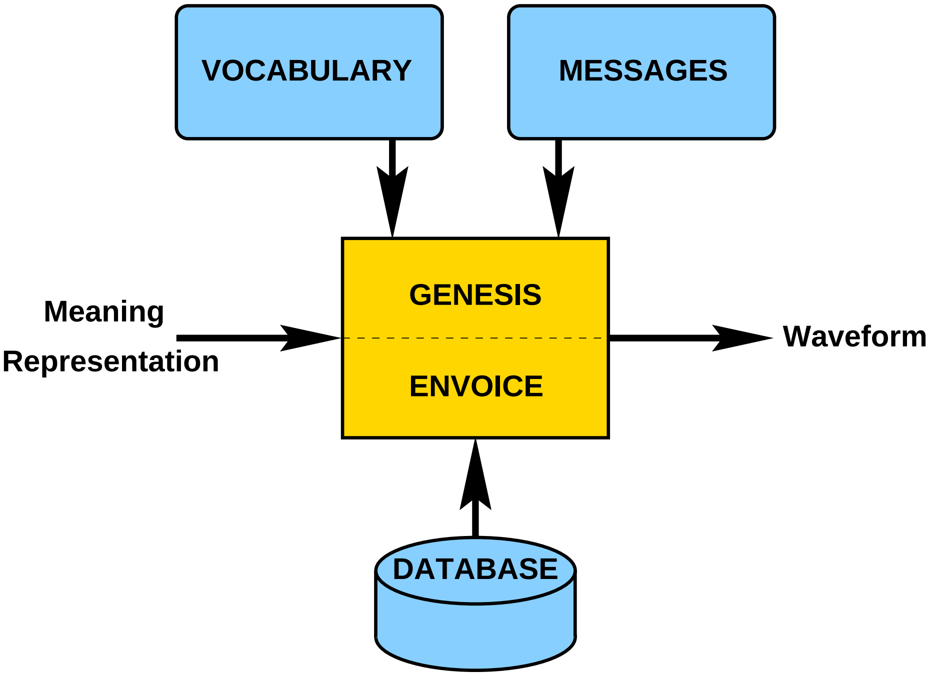


Figure 2-1: Word- and phrase-level concatenative synthesis architecture. GENESIS converts a meaning representation into a speech segment description using vocabulary items and message templates. The speech segment description assists ENVOICE in concatenating speech segments from the database to create the speech waveform output.

2.3 Response generation example

To first illustrate GENESIS at work, we present a simple WHEELS example of response generation. This will introduce each of the various components of the generation process from beginning to end. Afterwards, we shall extend the example to phrase-level concatenation.

In Figure 2-2, we see a meaning representation representing the query, “Show me all black cars costing less than $9000 with less than 50,000 miles.“ Various key-value pairs have been left out for clarity.

```
{c display
   :topic {q automobile
             :quantifier all
             :number "pl"
             :pred {p color
                      :topic "black" }
             :pred {p with_mileage
                      :pred {p less_than
                               :topic {q mileage_value
                                         :name 50000 } } }
             :pred {p costing
                      :pred {p less_than
                               :topic {q dollar_amount
                                         :name 9000 } } } }
}
```

Figure 2-2: Meaning representation for query to WHEELS system. This *display* clause contains an *automobile* topic which has three predicates: *color*, *with_mileage*, and *costing.*

After the WHEELS system searches the database of classified advertisements, it returns with a list of automobiles matching the search criterion. A possible database record in frame format is shown in Figure 2-3. Again, various key-value pairs have been left out for clarity.

To provide a response to the user, this database record is translated into a human-understandable format. When presented with the message templates *response1*, *response2*, and *response3* along with supplementary messages shown in Figure 2-4, the three following sentences can be generated.

The third ad is a 1996 black Acura Integra with 45,380 miles.
The price is 8,970 dollars.
Please call (404)399-7682.

We can see that certain values are simply filled in by GENESIS with record-specific information. As such, the set of responses with these three message templates is extremely constrained. It is these type of templates that can qualify as *carrier phrases.* Certain portions of the text are static, whereas other parts are dynamic and are filled in at generation time from the meaning representation. Before we describe carrier phrases in full, let us first repeat this example for phrase-level concatenative synthesis.

```
{
   :year 1996
   :color "black"
   :model "integra"
   :make "acura"
   :mileage 45380
   :price 8970
   :telno "404 399 7682"
   :nth_ad "third"
}
```

Figure 2-3: Meaning representation for database record.

```
response1    :NTH_AD_GEN :YEAR :CAR_FEA :CAR_NP :MILEAGE_GEN
:nth_ad_gen  The :NTH_AD ad is a
:car_fea     :COLOR :SIZE
:car_np      :MAKE :MODEL :CAR_TYPE
:mileage_gen with :MILEAGE miles
response2    The price is :PRICE dollars
response3    Please call :TELNO
```

Figure 2-4: Message templates from WHEELS.

2.4 Phrase-level concatenation example

In the previous generation example, a meaning representation was converted to a textual surface representation by GENESIS using a pre-defined set of vocabulary and messages. Now we will extend this example by modifying the vocabulary and messages to contain ENVOICE-specific annotation for specifying waveform segments for phrase-level concatenative synthesis. Afterwards, we will have had a complete view of the phrase-level concatenation system. Then, we shall describe the design behind the vocabulary and message templates, carrier phrases, and the speech database.

Before the WHEELS system passes the meaning representation to the phrase-level concatenative synthesizer, it slightly modifies the numbers for the automobile mileage, price, and telephone number. For example, the WHEELS system breaks up the mileage and price into the way it would be read out aloud. We shall explain the labeling of the telephone number later on. This modified meaning representation is seen in Figure 2-5.

```
{ ecad
   :domain "dWheels"
   :year 1996
   :color "black"
   :model "integra"
   :make "acura"
   :mileage {q number :1 40 :2 5000 :3 300 :4 and :5 80 }
   :price {q number :1 8000 :2 900 :3 and :4 70 }
   :telno {q telephone :1 "1_4" :2 "2_0" :3 "3_4"
     :4 "4_3" :5 "5_9" :6 "6_9" :7 "7_7" :8 "8_6" :9 "9_8" :0 "0_2" }
   :nth_ad "third" }
```

Figure 2-5: Modified meaning representation for database record. Mileage, price, and telephone number are 45,380 miles, 8,970 dollars, and (404)399-7682, respectively.

As the value structures of the mileage, price, and telephone number keys have changed, so must the messages that extract these key-value pairs. In the case of the telephone number, the new message simply needs to pull out the ten components in the correct order. Also, we begin to see the ENVOICE-specific annotations that describe the waveform segments. These slightly modified messages are seen in Figure 2-6.

```
response1    :NTH_AD_GEN :YEAR :CAR_FEA :CAR_NP :MILEAGE_GEN
:nth_ad_gen  [ wheels01 1000 9140 1.0 1.0 1.0 The ] :NTH_AD
               [ wheels01 12060 15780 1.0 1.0 1.0 ad is a ]
:year        :YEAR
:car_fea     :COLOR :SIZE
:car_np      :MAKE :MODEL :CAR_TYPE
:mileage_gen [ wheels01 43500 44660 1.0 1.0 1.0 with ] :MILEAGE
               [ wheels01 51664 59424 1.0 1.0 1.0 miles <pause2> ]
:mileage     :TOPIC
number       :1 :2 :3 :4 :5 :6
response2    [ wheels32 4000 11000 1.0 1.0 1.0 The price is ] :PRICE
               [ wheels32 22439 31776 1.0 1.0 1.0 dollars ]
:price       :TOPIC
response3    [ wheels34 4000 12000 1.0 1.0 1.0 Please call ] :TELNO
:telno       :TOPIC
telephone    :1 :2 :3 :4 :5 :6 :7 :8 :9 :0
```

Figure 2-6: Modified message templates from WHEELS.
Message templates have augmented with speech segment descriptions.

Within the messages above, we see embedded waveform segment descriptions delimited by square brackets. The fields within the brackets correspond to the utterance identification tag, start sample, end sample, duration scaling factor, pitch scaling factor, energy scaling factor, and the orthography. The scaling factors allow optional prosody modification; because they are all unity, there is no signal processing performed on the waveform.

The generation for this meaning representation is somewhat more involved for the mileage, price, and telephone number. The mileage and price both have *number* topics as their children. This *number* topic is evaluated by concatenating the subfields, *:1* through *:6*. The telephone number is a *telephone* topic evaluated by concatenating the subfields, *:1* through *:0*. Using these indexed subfields, we can guarantee the order of concatenation and record the vocabulary items accordingly.

As described before, phrase-level concatenation is a two-step process. The first step involves converting a meaning representation into a segment description. Then, the appropriate word- and phrase-level segments are spliced together with no signal processing. Now that the messages have been embedded with segment descriptions for the static portions, we also need to annotate the vocabulary entries with segment descriptions for the dynamic portions.

Figure 2-7 shows the segment description when the *response1* message template is applied to the meaning representation. Although we will describe carrier phrases in more depth in the next section, we can see that the utterance, *wheels01*, is the carrier phrase in this example. Function words and overall sentence structure are captured by *wheels01*. Note that a change in the utterance tag (left column) shows the concatenation of different utterances.

```
[ wheels01  1000  9140 1.0 1.0 1.0 The ]
[ wheels03  7300  9800 1.0 1.0 1.0 third ]
[ wheels01 12060 15780 1.0 1.0 1.0 ad is a ]
[ wheels13 15520 25920 1.0 1.0 1.0 1996 ]
[ wheels04 26920 30340 1.0 1.0 1.0 black ]
[ wheels45  7380 12340 1.0 1.0 1.0 acura ]
[ wheels45 13040 18264 1.0 1.0 1.0 integra ]
[ wheels01 43500 44660 1.0 1.0 1.0 with ]
[ wheels08 36220 39380 1.0 1.0 1.0 40 ]
[ wheels25 36460 42788 1.0 1.0 1.0 5000 ]
[ wheels27 38400 41160 1.0 1.0 1.0 3 ]
[ wheels32 19600 22439 1.0 1.0 1.0 hundred ]
[ wheels33 26160 27300 1.0 1.0 1.0 and ]
[ wheels04 41460 43860 1.0 1.0 1.0 80 ]
[ wheels01 51664 59424 1.0 1.0 1.0 miles ]
```

Figure 2-7: Segment description from *response1* message.

The final step of the two-step phrase-level concatenation process is carried out by the ENVOICE component. As input it takes in this segment description and outputs speech waveform output. For each segment, it loads the correct waveform, extracts a portion of speech, and places it into the synthesis buffer. Any optional prosody modification can be carried out at this point as specified by the pitch, duration, and energy scaling factors. However, generally no modification was performed in this work.

Now that we have seen an example thoroughly delineating the phrase-level concatenation system, we shall examine in more detail how the vocabulary and messages are designed and how the speech database is recorded. We will start by showing how a carrier phrase works.

2.5 Carrier phrase

As we saw from the examples, a carrier phrase contains static and dynamic portions. The static portion makes up the general meaning of the sentence, whereas the dynamic portion changes from realization to realization of the carrier phrase. By recording words and phrases in carrier phrases, we capture vocabulary items in a desired prosodic environment. These same carrier phrases can be used for synthesis templates, where we substitute words and phrases into the dynamic place-holders. In carrier phrases, co-articulatory constraints between words and phrases are not explicitly accounted for, although when we designed the prompts, we took care to try to minimize the number of different contexts. We will see how prosodic constraints can be more important than co-articulatory constraints at the word level and above.

Earlier we suggested that the GENESIS message templates could themselves be used as carrier phrases. This is true, because the messages templates fulfill the definition by containing static and dynamic portions. The message templates dictate the structure of the responses. Speech recorded in carrier phrases and ultimately destined for synthesis within the message templates will sound natural by virtue of matching the precise prosodic condition. For example, in the following two sentences, each of the underlined words can be replaced with an alternative spoken in the same context with little degradation in naturalness.

> The second ad is a 1960 large blue Toyota 4x4 with 100,000 miles.
> The fifth ad is a 1988 gray Nissan sedan with 70,000 miles.

2.6 Corpus design

In this section, we shall discuss corpus design and how the speech database is prepared. All the recording prompts are designed as carrier phrases to capture vocabulary in the precise prosodic environment in which they will be needed. In the case of cardinal numbers and telephone numbers, we shall see how permutation can be used to obtain digits in every possible position speakable.

2.6.1 Years

Recording in the precise prosodic environment is perhaps most concisely demonstrated in the recording of years. Essentially, a year is composed of a century, a decade, and a digit for the exact year. Because the century, decade, and digit are always spoken in the same order and manner, they can be freely interchanged with little degradation in naturalness. To capture the prosody, we must record the centuries, decades, and digits in the exact position that they will be used. In Figure 2-8, we see all the years that need to be recorded to cover the latter half of the 20th century.

We conjecture that after 1950 is recorded, 1960 is not needed in its entirety; only

```
1950 1960 1970 1980 1990
1981 1982 1983 1984 1985 1986 1987 1988 1989
```

Figure 2-8: Years needed for coverage of latter half of 20th century.

the decade portion is required as the century has been recorded once in its correct position. The choice of decade used in the recording of the digits can then be totally arbitrary.

2.6.2 Ordinal numbers

In Figure 2-9, we see the phrases that are recorded for coverage of the ordinal numbers. These phrases also happen to be the very utterances that form a portion of the actual *WHEELS* phrase-level corpus.

```
The first ad is a 1950 medium-size white Volvo station wagon
    with 200,000 miles.
The second ad is a 1960 large blue Toyota 4x4 with 100,000 miles.
The third ad is a 1970 small red Porsche convertible with
    90,000 miles.
The fourth ad is a 1980 compact black Honda hatchback with
    80,000 miles.
The fifth ad is a 1988 gray Nissan sedan with 70,000 miles.
The sixth ad is a 1989 tan Ford sport utility vehicle with
    60,000 miles.
The seventh ad is a 1990 green Volkswagen van with 50,000 miles.
The eighth ad is a 1991 silver Chrysler minivan with 40,000 miles.
The ninth ad is a 1992 gold GMC truck with 30,000 miles.
```

Figure 2-9: Phrases to record for coverage of ordinal numbers.

2.6.3 Cardinal numbers

For reporting numbers, such as car mileage, it is necessary to capture all the possible combinations of digits and units (tens, hundreds, thousands). It is obvious that all ten digits are needed in a word-final position. Recording all the possible enumerations will lead to full coverage. For example, in *10,450* and *20,380*, the "ten" and "twenty" that is spoken can be interchanged with little degradation in naturalness.

2.6.4 Telephone numbers

In the United States, phone numbers contain three portions: a three-digit area code, a three-digit prefix/exchange, and a four-digit number. We hypothesized that by recording all the digits in each of these $3 + 3 + 4 = 10$ positions, prosodic conditions could be covered. This can be achieved by recording ten fictional telephone numbers where each telephone number is just a shifted version of the one before as shown below:

> Please call 012 345 6789.
> Please call 123 456 7890.
> Please call 234 567 8901.
> Please call 345 678 9012.
> Please call 456 789 0123.
> Please call 567 890 1234.
> Please call 678 901 2345.
> Please call 789 012 3456.
> Please call 890 123 4567.
> Please call 901 234 5678.

The example of phone numbers best illustrates how prosodic constraints can be more important than co-articulatory constraints in phrase-level concatenation. This type of phrase-level recording only takes into consideration prosodic constraints across words, and not co-articulatory constraints. For example, in comparing a *34* and *38* sequence, we find that formant motion in the vowel of *3* can differ. Even though arbitrary telephone numbers generated from this ten-sentence corpus can sound natural, these ten permutations that capture all the prosodic environments do not capture all the co-articulatory environments. An upper bound would dictate the recording of the 10 digits in the 8 internal positions with 10^2 contexts to the left and right and in the 2 external positions with 10 contexts on only one side. Compression could possibly be achieved by collapsing co-articulation constraints and by finding shared prosodic conditions.

2.6.5 Other car attributes

Once again, we can apply the concept of prosodic environment to reduce the amount of recording. In recording car makes and models, we really only need each make recorded once followed by an arbitrary model. We hypothesized that the rest of the models for that make can be recorded in isolation. Preliminary experimentation showed that using makes recorded in isolation can produce natural-sounding synthesis.

2.7 Vocabulary and message preparation

With the above in mind, we can put together a set of prompts for a human to speak in a recording session. These prompts must not only cover the carrier phrases, but also the multitude of changing variables that can be filled into the carrier phrases. Although we explained how to record each of the word types (years, cardinal numbers, telephone numbers, etc.) in isolation, we can combine the concepts and design verbose recording prompts with the structure of the three WHEELS message templates that provide maximum coverage in a compact manner.

After the speech corpus is recorded, a final step of preparation remains before it can be used for phrase-level concatenation. A GENESIS vocabulary file must be created to serve as a mapping from the values in the meaning representation to waveform segment descriptions. First, the orthography is transcribed. Then, the orthography is formatted as waveform segment descriptions within GENESIS vocabulary entries. Example vocabulary entries can be seen in Figure 2-10.

```
wagon "[ wheels01 40920 43500 1.0 1.0 1.0 wagon ]"
third "[ wheels03  7300  9800 1.0 1.0 1.0 third ]"
black "[ wheels04 26920 30340 1.0 1.0 1.0 black ]"
```

Figure 2-10: Example vocabulary items.

Each entry in the vocabulary is mapped to a segment in the database as described by a segment description as shown in Figure 2-10. This vocabulary lookup feature allows for very powerful mappings which can achieve multi-lingual generation (e.g., in German, third → dritte), for example. In the case of concatenative speech synthesis, we are using this ability to pass on annotations to the phrase-level concatenative synthesizer. As mentioned earlier, each segment description encodes the utterance identification tag, start sample, end sample, duration scaling factor, pitch scaling factor, energy scaling factor, and the orthography.

Now we are ready to explain the labeling of the telephone number in the modified meaning representation displayed within Figure 2-5. As the above section on telephone numbers showed, the fictional telephone number, (012)345-6789, is repeatedly shifted nine times for a total of ten telephone numbers. This covers each of the ten digits in each of the ten positions. For example, in Figure 2-11, we see the third telephone number represented as ten vocabulary entries.

```
1_2 "[ wheels36 10620 13340 1.0 1.0 1.0 2 ]"
2_3 "[ wheels36 13340 15700 1.0 1.0 1.0 3 ]"
3_4 "[ wheels36 15700 20600 1.0 1.0 1.0 4 ]"
4_5 "[ wheels36 20600 24080 1.0 1.0 1.0 5 ]"
5_6 "[ wheels36 24080 27220 1.0 1.0 1.0 6 ]"
6_7 "[ wheels36 27220 31140 1.0 1.0 1.0 7 ]"
7_8 "[ wheels36 31140 34552 1.0 1.0 1.0 8 ]"
8_9 "[ wheels36 34552 38512 1.0 1.0 1.0 9 ]"
9_0 "[ wheels36 38512 42211 1.0 1.0 1.0 0 ]"
0_1 "[ wheels36 42211 45031 1.0 1.0 1.0 1 ]"
```

Figure 2-11: Example telephone number.

The values in the modified meaning representation from Figure 2-5 are keyed such that the *response3* message in Figure 2-6 extracts the key-value pairs in the correct order. So, both the message and vocabulary entries have been encoded with positional information. This guarantees that the digits will be synthesized in the precise prosodic environment in which they were recorded.

2.8 Discussion

In the framework described above, GENESIS is able to generate a waveform segment description from a meaning representation, vocabulary, and message templates. The annotations guide ENVOICE in selecting speech segments from the corpus by providing a waveform identification tag, start time, and end time. ENVOICE is then able to take the annotated generation output, and perform simple concatenative synthesis at a phrase level with no signal processing.

Utilizing this process of careful corpus design for response generation, it is possible to achieve very natural concatenated speech in a constrained application domain. The WHEELS system is an example of a conversational language system that makes use of this phrase-level synthesis scheme. System users have found the human voice synthesis preferable over other speech synthesizers such as DECTalk.

The corpus design involves the recording of carrier phrases and constituent words and phrases. The design of the carrier phrases follows from the message templates of the conversational language system. Once the corpus is recorded, we have a static vocabulary (zero vocabulary flexibility) and moderate sentence flexibility (carrier phrases can be instantiated with different vocabulary in a constrained order). However, depending on the type of vocabulary, there could potentially be a large of amount of recording needed for absolute full coverage.

For example, in an air travel domain we shall see later on, the source and destination cities need to be spoken as part of the response. In the United States, there are around 23,000 possible cities. Another example lies in the WHEELS domain where car models are numerous for each automobile manufacturer and new models may be introduced each year. The restaurant domain is another example where new names are arising on a regular basis.

Thus, city names, car makes, car models, and more generally, proper names, are types of words that can potentially have large set sizes which may also grow as time goes on. While brute-force methods would dictate the recording of every word, we decided to seek out methods that concatenate sub-word units for the synthesis of arbitrary proper names.

Sub-word concatenation could be used in conjunction with a mitigating strategy of recording words that frequently occur. For example, in WHEELS, only car makes and models that are common would be recorded, and less common makes and models would rely on sub-word concatenation.

The new word problem manifests itself in both speech recognition and synthesis; especially so in the larger space of proper names. While the strategy we will develop in chapters to come is applicable to the synthesis of all words - not just proper names - we focused our research on city names.

Finally, phrase-level recording only attempts to capture prosodic constraints, and does not explicitly capture co-articulatory constraints. We found that in speech segments at levels higher than a word, prosodic constraints tend to be more important that co-articulatory constraints. However, as we examine sub-word units in the following chapters, we will see that matching the co-articulatory environment can be more important.

Chapter 3

Perceptual experiments

The phrase-level concatenation framework introduced in the previous chapter splices together phrases at word boundaries by implicitly matching prosodic context, with no regard for co-articulatory constraints. The sub-word concatenation framework we begin to examine in this chapter performs in the opposite manner: it concentrates on matching co-articulatory context first, with prosody being a secondary concern.

In this chapter, we perform several perceptual experiments in which we attempt to learn what units are appropriate for concatenative synthesis, and how well these units sound as an ensemble when concatenated together to form new words. We also try to discover what contexts may be substituted for other contexts with little or no perceptual distortions. These tests not only describe where the unit boundaries should lie, but also how much contextual information is required.

Ultimately, the knowledge gained in this chapter will help us lay out a foundation for selecting units from a speech database. Individually, the units must match the specific units required to form new words. As an ensemble, they must connect together to form natural-sounding transitions from unit to unit. For vowels for example, these two criterion are equivalent to determining which contexts have the same effect on formant frequencies, and what units need to be selected to minimize formant discontinuities at the concatenation boundaries. We call these two constraints the *unit* and *transition* criterion.

Some of the experiments provide empirical support for either the unit, or the transition criterions; some simultaneously support both. In all cases, we try to provide both positive and negative examples, or other evidence, for each of the hypotheses we test. The series of experiments build up evidence in a cumulative fashion by starting with experiments performed on units as small as phones and leading into experiments involving larger-sized units.

3.1 Speech production background

As depicted in Figure 3-1 [23], a simple model of human speech production has a switched source representing the glottal and sub-glottal system flowing through a filter representing the vocal tract. In this simple model, the two mutually exclusive modes of production are controlled by voiced and unvoiced excitations.

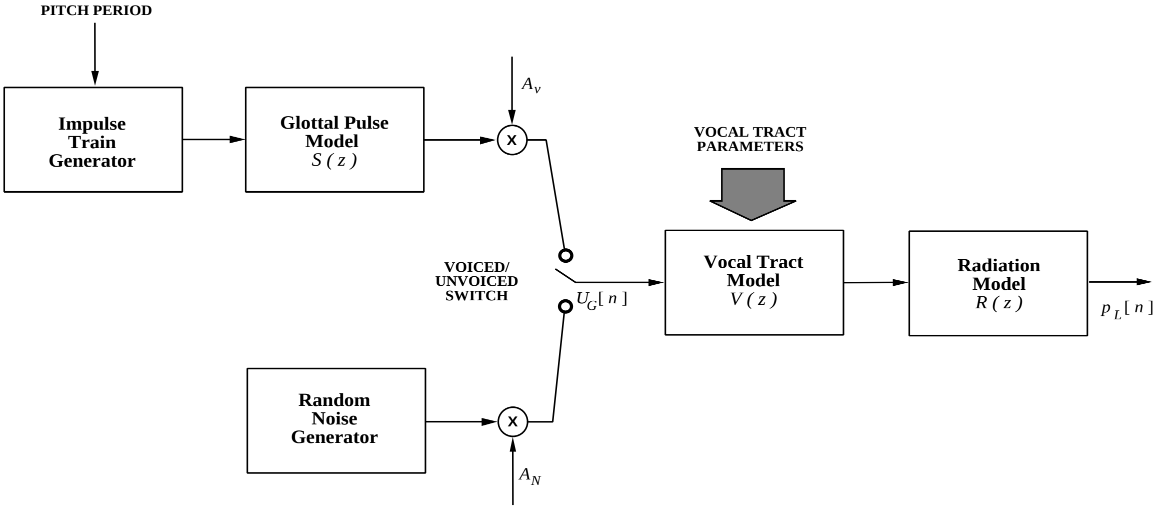


Figure 3-1: Source-filter model of human speech production [23].

One of the first hypotheses for transition points was where the source changes.

Changes in source (e.g., voiced-unvoiced transitions) typically result in significant spectral changes [8]. We hypothesized that because of this, a splice might not be as perceptible at this point, in comparison to other places. Should the speech signal be broken between two voiced regions, it would be important to ensure formant continuity at the splice boundary. In breaking the speech signal between two regions where at least one is an unvoiced region, it will be less important to maintain formant continuity. Note that this type of concatenation differs from diphone synthesis [14], which usually transitions from one unit to another in the middle of a phone.

3.2 CVC experiments

3.2.1 Place of articulation

The first series of experiments deal with the substitution of what will be the smallest unit, a phone. Specifically, the vowel from within a consonant-vowel-consonant (CVC) sequence will be replaced by another vowel from another CVC sequence.

In the top of Figure 3-2, we see the spectrogram of the city name, “Boston”, which has the phonemes: /b ɔ s t ɪ n/. In the bottom of Figure 3-2, we see “Boston” again, but this time /ɔ/ has been taken from “bossed”, which has the phonemes: /b ɔ s t/.

Since we have kept the context for /ɔ/ constant, we can examine how this example supports the action of splicing. We can see that the spectrograms look very similar. Perceptually the splicing is not noticeable. We found this effect to hold when the consonants are stops, fricatives, as well as affricates. The contextual information required for this type of splicing is the place of articulation to the left and right of the vowel. We created three classes: labial, alveolar/palatal/dental, and velar. We found alveolar (t d s z), dental (θ ð), and palatal (š ž č ǰ) contexts to have similar effects on formant frequency locations, and produced natural-sounding synthesis.

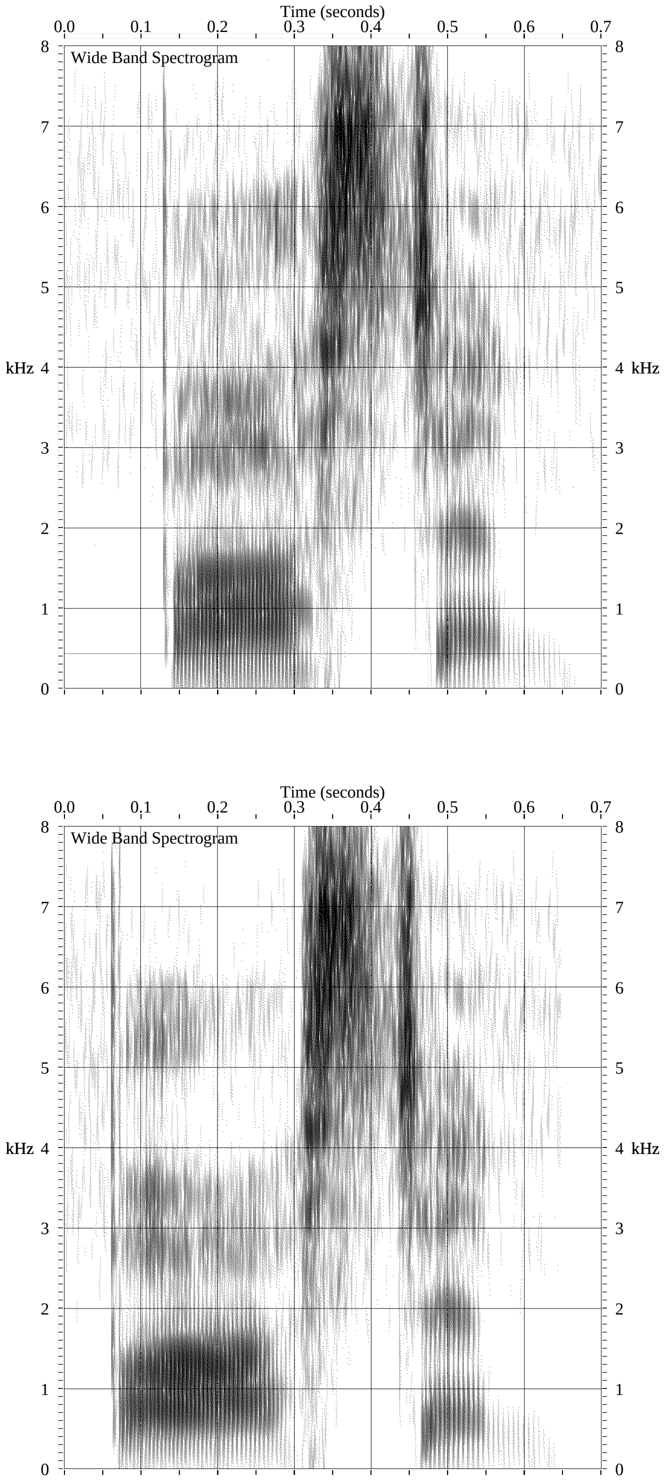


Figure 3-2: Spectrograms of actual and synthetic "Boston." Synthetic version has /ɔ/ taken from "bossed."

3.2.2 Effects of voicing

Because the above experiment has shown a case where splicing can be performed well, we can begin to vary some parameters. Namely, we shall take the same example and vary the voicing dimension to determine what can serve well as a unit criterion.

In the top of Figure 3-3, we see the spectrogram for the word "paucity" which has the phones: /p ɔ s ɪ t i^y/. In the bottom of Figure 3-3, we once again substitute the /ɔ/ phone in "Boston". This time, the voicing dimension of the stop consonant on the left side of /ɔ/ has changed.

Because the stop is unvoiced, we observe that the voice onset time (VOT) in Figure 3-3 is noticeably longer and that part of the /ɔ/ is devoiced in "paucity", so that there is less formant transition in the voiced part of the vowel. Despite the fact that there is less formant motion in the voiced portion of the vowel, perceptual listening indicated that this was a secondary effect, and that overall the splicing still sounded natural. Finally, we note that the duration of the vowel was also shorter. This may affect rhythm in poly-syllabic words if incorrectly used, but this could be possibly be dealt with by duration modification methods.

In the previous experiment we were able to isolate where a good splicing boundary is. Building on that, we have found that the place of articulation on the left side of a vowel in a CVC sequence is important, and that the voicing dimension is not as important.

By symmetry arguments, this principle can be extended to the right side as well. Thus, the piece of knowledge we have acquired towards our definition of a unit criterion is that in a CVC sequence, a minimal description for the context of the vowel is the place of articulation on both sides, a trigram context.

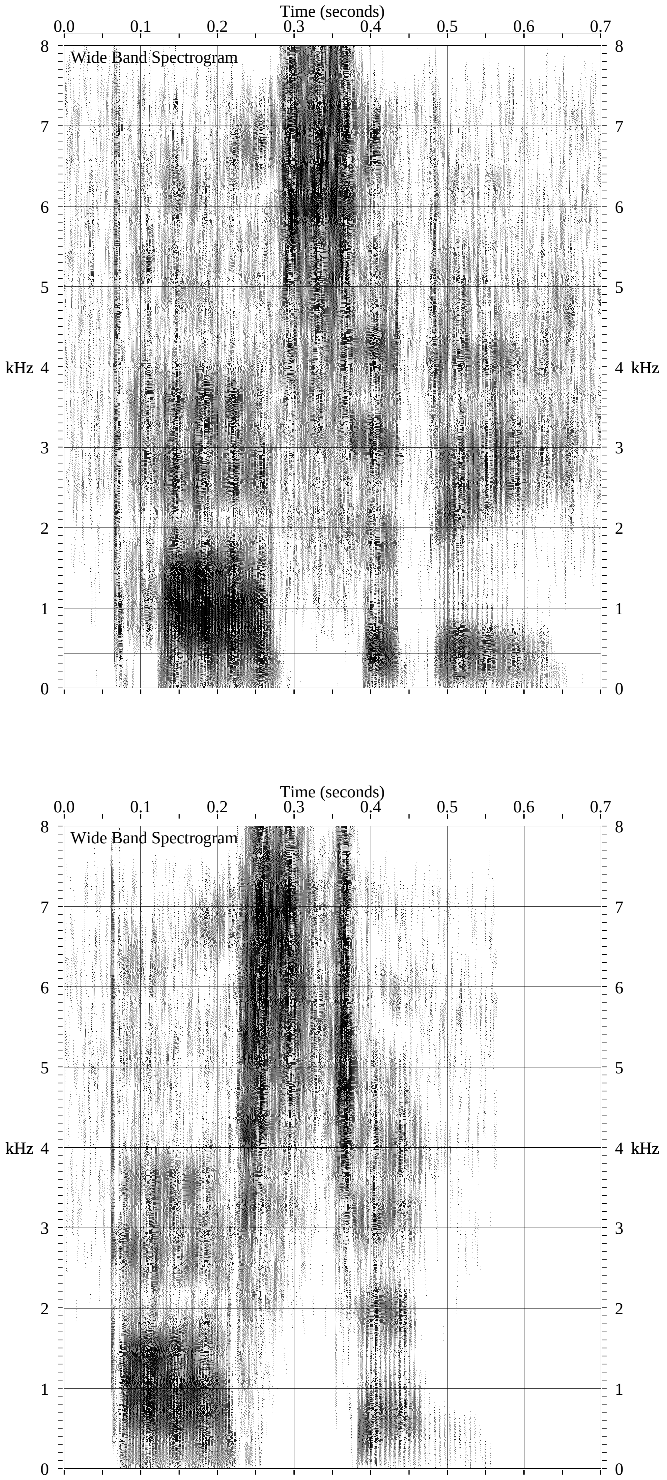


Figure 3-3: Spectrograms of "paucity" and synthetic "Boston." Spectrogram of "Boston" with /ɔ/ from "paucity."

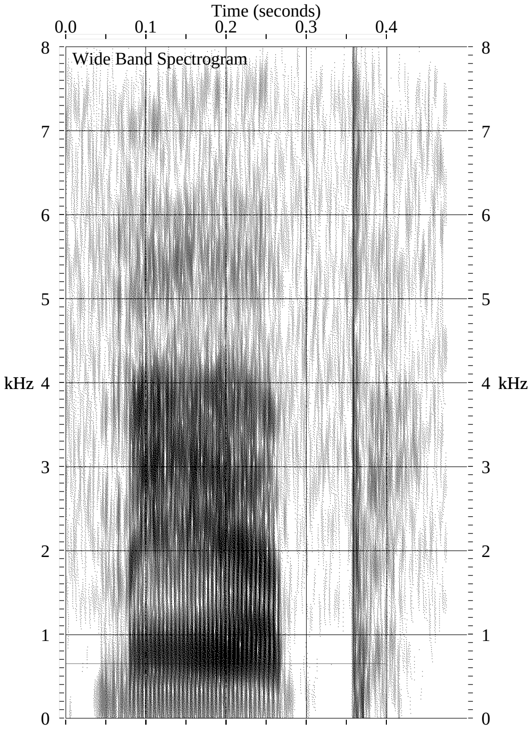
3.2.3 Nasal Consonant CVC

As a variation on the previous experiment type, we next examined the substitution of a vowel from within a CVC sequence, where either, or both consonants can be nasal consonants. However, we kept the place of articulation of the consonants constant throughout the experiment. This experiment will help us to understand the phenomenon of nasalization.

In Figures 3-4 and 3-5, we see the spectrograms for three of the four combinations possible when varying the nasal dimension of a CVC sequence where both consonants are labial ("map", "pam", "pap", and a synthetic "pap" where the /æ/ is taken from "map"). From listening, it was clear that the /æ/ in the first two cases are nasalized to some extent. For example, when the /æ/ from "map" is substituted into "pap" as shown in Figure 3-5, listening reveals a nasalized vowel, which did not sound natural.

Nasalization of a vowel occurs when a consonant to either side is a nasal. Hence, we will not be able to dismiss it from the unit criterion definition as we did with voicing. The knowledge of the nasalization dimension is indeed required in the contextual information. Continuity in vowel nasalization and in the nasal murmur across the vowel-nasal boundary is important to maintain [12]. We found this effect to be stronger for vowel-nasal sequences than for nasal-vowel sequences, possibly because anticipatory nasalization is a stronger effect in English.

One last comment on nasal consonants pertains to their splice boundaries. In Figure 3-4, we see the closures in the oral cavity marked by sudden changes in energy as well as formant structure. Because this event is not gradual, but abrupt, it is only natural to allow splicing at such a boundary. This contributes to our knowledge of what defines the transition criterion.



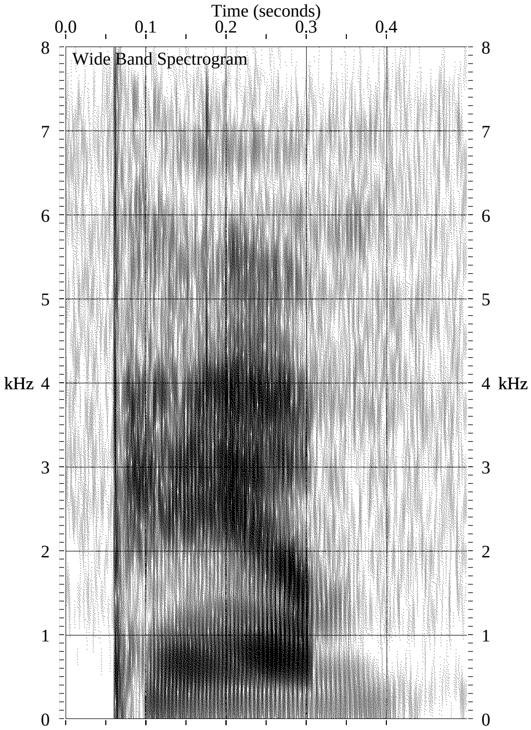


Figure 3-4: Spectrograms of “map” and “pam.”

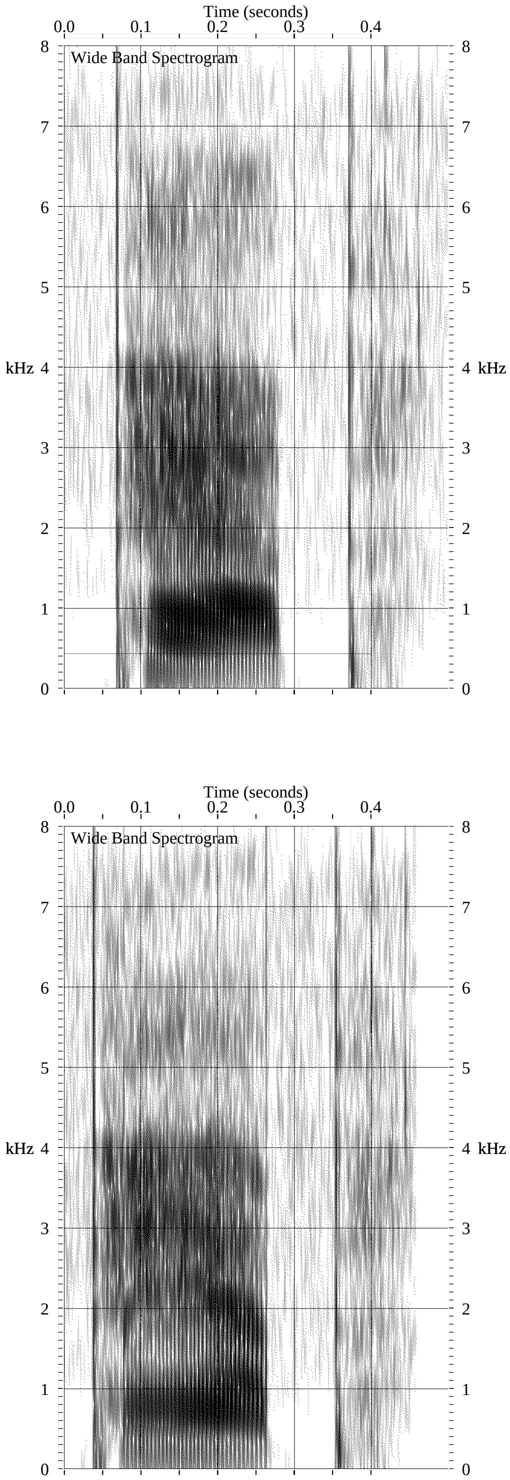


Figure 3-5: Spectrograms of actual and synthetic “pap.” Synthetic “pap” has /æ/ taken from “map.”

3.3 Discussion

The experiments in this chapter have illustrated the role that co-articulatory constraints might play in selecting and concatenating units. The respective unit and transition criterions describe these two processes. As these constraints are met, the co-articulation is improved. Mainly we have found the place of articulation and nasalization to be the main contextual constraints.

The results of these perceptual experiments will be used in the next chapter for determining the unit inventory. Since we have found boundaries between vowels and consonants to be places where splicing can be performed while maintaining naturalness, we hypothesize it will be important to keep vowel and semivowel sequences contiguous. Also, the evidence supporting the decoupling of the unit and transition criterions and the definitions thereof will be used in the development of the search framework.

Chapter 4

Design of synthesis units

4.1 Introduction

As discussed in Chapter 3, we conducted many experiments that investigated how segments of speech can be substituted without noticeable perceptual distortions. Working with a minimal unit of a phoneme, we learned what contextual constraints need to be maintained to perform this type of concatenation. Looking at places of source change in speech production, we also found additional concatenation points.

In this chapter, the various principles learned from the perceptual experiments are used to enumerate a set of synthesis units for concatenative synthesis of non-foreign English words. As part of the study, a set of words are automatically generated to serve as a sub-word corpus that compactly covers the co-articulatory inventory required for unconstrained synthesis.

Since the space of units can grow large as context is added [31], the use of place of articulation classes can help to reduce the number of contexts. To combat the growth of synthesis unit length, we will take full advantage of concatenation splice points to permit breaking the speech signal into shorter fragments. By fixing the splice boundaries, we define the units themselves.

The perceptual experiments in the previous chapter helped to identify what a minimal concatenation unit can be, and what type of contextual information is needed to maintain natural-sounding synthesis. This trigram context incorporated information about only the left and right side is incorporated, and led to the formation of the unit criterion. Also, we learned where transition points could be placed without gross perceptual distortions.

In this chapter, a lexicon of non-foreign English words is used to perform an analysis of multi-phoneme sequences. This will define an upper bound on the unit inventory required for unconstrained synthesis. We will make compromises in the maximum length of multi-phoneme sequences to cover in order to keep the sub-word corpus appreciably compact. This is because most longer multi-phoneme sequences occur in only a small number of the words within the lexicon.

4.2 Analysis of English language

To determine the units required for synthesis, we made use of a 90,000-word lexicon from the Linguistic Data Consortium called the *COMLEX English Pronunciation Dictionary* [20], commonly referred to as PRONLEX. We limited our analysis to the non-foreign subset of PRONLEX containing approximately 68,000 words.

The analysis of multi-phoneme sequences focused on contiguous vowel and semivowel sequences in PRONLEX, since they comprise the majority of the synthesis units. Consonant sequences were ignored, since we believed we would have adequate coverage from any recorded corpus. Finally, in the interest of reducing the unit space size, lexical stress was ignored. Although lexical stress is important for natural-sounding synthesis, we shall see in the next chapter how special labels for reduced vowels can be created as the lexical stress is removed.

After focusing the analysis on vowel and semivowel sequences and discarding lexical stress, we proceeded to enumerate all such sequences contained in the 68,000 words of the non-foreign subset of PRONLEX. We allowed these vowels and semivowels to chain together, when not separated by a consonant, to produce even larger units. The result of this analysis is shown in Table 4-1. Note that the cumulative coverage of vowel and semivowel sequences with lengths of 1, 2, and 3, increases from 63.3%, to 89.0%, and to 97.1%, respectively.

Length	Example	# of units	# of occurrences	% cumulative
1	ɪ	16	100513	63.3
2	rɪ	167	40778	89.0
3	ərɪ	484	12916	97.1
4	yəle^{y}	637	3392	99.3
5	o^{w}ri^{y}əl	312	906	99.8
6	yələrɑ^{y}	80	226	100.0
7	ærəlɛlɪ	13	21	100.0
Total		1709	158752	100.0

Table 4-1: Analysis of vowel and semivowel sequences in PRONLEX.

The vowel and semivowel sequences of length 1 are just the 16 vowels in English. However, we wanted to know the structure of these sequences of length 2, and higher. Table 4-2 shows the breakup of the vowel and semivowel sequences of length 2, where S denotes semi-vowel, and V denotes vowel. Also, individual semi-vowels are reported on. It is interesting to note that the SV and VS sets are close to their theoretical maximum limits of 64 and 32, respectively. However, the VV set is far from the theoretical maximum of 256.

A similar analysis of vowel and semivowel sequences of length 3 was performed on a coarser level as shown in Table 4-3. The two figures that are significant are the counts for VSV and SVS. We conjecture as future research that it is possible to manufacture VSV from constituent VS and SV components. The VSV set comprises 80.5% of all the occurrences of sequences of length 3. Thus, with total coverage of such sequences of length 2, many of the sequences of length 3 can be manufactured as well. As SVS units imply entire syllables, it may be better to record them as a whole unit.

Structure	Finer structure	# of units	# of occurrences
SV		62	25416
	rV	15	13535
	lV	16	7700
	wV	16	2764
	yV	15	1417
VS		29	12147
	Vl	16	7226
	Vr	13	4921
VV		76	3215
Total		167	40778

Table 4-2: Analysis of vowel and semivowel sequences of length 2 in PRONLEX.

Structure	# of occurrences
VSV	10404
SVV	1163
SVS	1145
VVS	174
VVV	24
VSS	6
Total	12916

Table 4-3: Analysis of vowel and semivowel sequences of length 3 in PRONLEX.

4.3 Unit inventory

With the units from the PRONLEX analysis, a unit inventory can be prepared. In the first stage, the vowel and semivowel sequences are expanded with trigram consonant contextual information. Silence is used as one of the contexts to mark word boundaries so synthesis can be performed from a corpus of isolated words. The result of this tabulation for these sequences is shown in Table 4-4.

With 20 consonants and silence, the theoretical expansion of these sets is $(20+1)^2 = 441$ when compared to Table 4-1. Fortunately, the actual expansion is not as large as demonstrated in Table 4-5. Next, we compressed the left and right consonant context using the seven contextual classes learned from the perceptual experiments: labial, alveolar/dental, velar, m, n, ŋ, and silence. This reduced the number of units as can be seen from Table 4-6.

Length	# of units	# of occurrences
1	2970	100513
2	4814	40778
3	3883	12916
4	1643	3392
5	515	906
6	110	226
7	16	21
Total	13951	158752

Table 4-4: Analysis of vowel and semivowel sequences with consonant/silence context in PRONLEX.

Length	# of units	# of units with context	Expansion factor
1	16	2970	185.6
2	167	4814	28.8
3	484	3883	8.0
4	637	1643	2.6
5	312	515	1.7
6	80	110	1.4
7	13	16	1.2
Total	1709	13951	8.2

Table 4-5: Expansion of sonorant units when context is applied.

Length	# of units with manner context	# of occurrences
1	541	100513
2	1817	40778
3	2343	12916
4	1342	3392
5	463	906
6	97	226
7	16	21
Total	6619	158752

Table 4-6: Analysis of sonorant units with manner context.

With 7 manner classes, the theoretical expansion of these sets is $7^2 = 49$ starting from the baseline vowel and semivowel sequences in Table 4-1. However, again this expansion is not as large as shown in Table 4-7.

Length	# of units with context	# of units with manner context	Expansion factor
1	16	541	33.8
2	167	1817	10.9
3	484	2343	4.8
4	637	1342	2.1
5	312	463	1.5
6	80	97	1.2
7	13	16	1.2
Total	1709	6619	3.9

Table 4-7: Expansion of sonorant units with context when manner classes applied.

4.4 Unit coverage

Now is the time to make compromises as alluded to earlier. To summarize the results of the analysis, we display a cross comparison across a few chosen dimensions in Table 4-8. We must choose the maximum vowel and semivowel sequence length to cover. This will choose an operating point on the operating characteristic curve. Although with a sequence length of 3 we can achieve 97.1% unit coverage and 93.3% word coverage, we choose to only cover up to sequences of length 2.

To synthesize any of the 68,000 non-foreign words in the English language, we need the 6,619 vowel and semivowel sequences with manner context as described above and the consonant units with manner context which were not enumerated, but assumed to be adequately covered. Choosing the operating point of 2, we need only cover $541 + 1817 = 2358$ unique sequences. A brute-force approach could be used where 2,358 words are recorded, where each covers a unit. Clearly, this can and should be done in a more efficient manner. The next subsection outlines a selection algorithm that is used to compactly cover the 2,358 unique sequences.

Sequence length	1	2	3	4	5	6-7
# of units	541	1817	2343	1342	463	113
# of occurrences	100513	40778	12916	3392	906	247
% unit coverage	63.3	89.0	97.1	99.3	99.8	100.0
% word coverage	29.8	75.2	93.3	98.3	99.6	100.0

Table 4-8: Comparison of vowel and semivowel sequences and associated word coverage.

4.4.1 Prompt selection algorithm

To prepare the sub-word corpus covering the 2,358 unique vowel and semivowel sequences, we used an automatic algorithm to select a set of words to record. To synthesize any word from the 68,000-word non-foreign subset of PRONLEX, this sub-word corpus can then be searched for an optimal sequence given a pronunciation. Words containing sequences of at most length 2 will be guaranteed to have excellent co-articulatory continuity. Synthesis of sequences of more than length 2 will fall back to using shorter constituents.

The automatic algorithm returns a set of prompts to record given a set of units to cover and a set of words to choose from. The prompt selection algorithm that is used selects the next best word to incrementally cover the most infrequent units remaining to be covered without providing already covered units [18, 19]. This process is iteratively performed until all the units have been covered.

We present the algorithm below with two concepts defined: A merit score of a word is the sum of the reciprocal of the number of occurrences of each unit. This will favor first selecting words containing rarer units and minimize overall duplication. When a word is fully covered, its merit score is set to zero. A demerit score of word is the number of redundant units it provides.

- Select the word with the highest merit score: $s_{merit} = \sum_i \frac{1}{N_i}$
    - Break a tie by selecting the word with the fewest demerits

  - Break a further tie by randomly selecting among the shortest words

- Update the merit and demerit scores of the remaining words

4.4.2 Recording prompts

When this prompt selection algorithm is applied to the 68,000 non-foreign English words to cover 2,358 vowel and semivowel sequences, a total of 1,604 words are selected. The list can be found in Appendix A.

Because ties can be broken randomly in the prompt selection algorithm as presented here, this list of 1,604 words is just one possible realization. Clearly, the process of breaking ties can be modified to produce deterministic results.

4.5 Discussion

This chapter has focused on tabulating the vowel and semivowel sequences present in the English language. This analysis was carried out on the 68,000 non-foreign words in the PRONLEX lexicon. The expansion effects of adding context and manner context to the unit labels have been calculated.

As a first-order approximation, we chose to ignore multi-consonant sequences. In Table 4-9, we see enumerated all the 74 word-initial consonant clusters in English. The clusters with semivowels are not pertinent to our unit inventory as we are working with vowel and semivowel sequences. We see that this number is far surpassed by the 6,619 vowel and semivowel sequences in the sub-word corpus. It can be argued by probability that covering the larger set of vowel and semivowel sequences will adequately cover the significantly smaller set of consonant sequences. There also exists a set of word-final consonant clusters that may have different acoustic realization than their word-initial counterparts (e.g., affix clusters). Upon considering only consonant-consonant sequences in Table 4-9, we find only five such sequences: /sf/, /sk/, /sp/, /st/, and /ts/. Because the boundaries between an /s/ and the three unvoiced stops can serve as splice points, this leaves only /sf/ and /ts/ which must be recorded as an entire unit. It is more important to cover consonants in different co-articulatory positions. Furthermore, it may be possible to compress the contexts around consonants as well (e.g., /sm/ and /sw/ share similar labial co-articulation in the /s/.)

-	of	hy	human	sf	sphere	tr	true
b	be	ǰ	just	sk	school	ts	tsunami
bl	black	k	can	skl	sclerosis	tw	twenty
br	bring	kl	class	skr	screen	ty	tuesday
by	beauty	kr	cross	skw	square	θ	thief
č	child	kw	quite	sky	skewer	θr	through
d	do	ky	curious	sl	slow	θw	thwart
dr	drive	l	like	sm	small	ð	the
dw	dwell	m	more	sn	snake	v	very
f	for	mw	moire	sp	special	vw	voyager
fl	floor	my	music	spl	split	vy	view
fr	from	n	not	spr	spring	w	was
fy	few	p	people	spy	spurious	y	you
g	good	pl	place	st	state	z	zero
gl	glass	pr	price	str	street	zl	zloty
gr	great	pw	pueblo	sw	sweet	zw	zweiback
gw	guava	py	pure	š	she	ž	genre
h	he	r	right	šr	shrewd		
hw	which	s	so	t	to		

Table 4-9: The 74 word-initial consonants in English.

Chapter 5

Search

5.1 Introduction

In the past few chapters, we have looked at the various components required to perform sub-word unit concatenative synthesis. In Chapter 3, we learned through experiments where the speech signal can be broken and spliced with little perceptual distortion. We also learned where the signal should not be broken and used knowledge from both the positive and negative examples to design constraints. These constraints were used in Chapter 4 in a sub-word analysis of an English lexicon. We enumerated a set of sub-word units required for the synthesis of any word from that same lexicon. Using an iterative prompt selection algorithm, we designed a corpus which would provide compact coverage of the sub-word unit set.

We are now ready to introduce the final component of the sub-word unit synthesis system, the unit selection search algorithm. This algorithm will provide an automatic means to select a sequence of sub-word units from a speech database given an input pronunciation. The units that arise out of the selection must try to match as best as possible the co-articulation environment dictated by the pronunciation. They must also line up well in unison. Finally, because the quality of synthesis tends to decrease as the number of concatenations increase [4], it is important to maximize the size and the contiguity of speech segments that are spliced together.

As we search over a database of phonemes, maximizing the size and contiguity of speech segments will encourage the selection of multi-phone sequences. This provides an elegant form of back-off in the case when the sub-word unit is not found. The search algorithm will select as many phones in sequence as possible. If the speech database were only marked with sub-word unit boundaries, then this degree of freedom would not be available. However, since the corpus was recorded with maximal coverage of sub-word units in mind, those very sub-word units must exist within the speech database. When we refer to searching over a speech database of sub-word units, in actuality we are referring to searching over a speech database of phones that produce sub-word units as multi-phone sequences when contiguity is maximized.

5.2 Viterbi search algorithm

As a first step, we decide on a search algorithm to adopt for unit selection. We choose the *Viterbi search* [9] for its ability to search a graph in a time-synchronous manner and its excellent pruning characteristics lent by its dynamic programming formulation. After the Viterbi search is complete, the Viterbi path is obtained from the backtrace through the graph. This path is guaranteed to be optimal over the entire graph. As the Viterbi search is carried out, a cost is assigned to each node of the Viterbi lattice. This represents the lowest cost attainable out of all the best possible paths from the beginning time point through any given node.

5.3 Search metric

While the search algorithm determines how a graph is searched, what ultimately will determine the overall synthesis quality, intelligibility, and naturalness is the metric used for unit selection. The search metric must account for all of the speech knowledge about a language necessary for concatenative speech synthesis as encoded by the researcher. The search metric will allow the search algorithm to seek out units that individually match the input specification well, that connect well as an ensemble, and that are maximally contiguous.

The first two of the three desirable attributes of a search metric were the primary driving force behind the perceptual experiments in Chapter 3. Since these two criterion are decoupled, they can be separately considered; the cost function is split into a unit cost function and a transition cost function [17]. Picturing units as nodes in a unit graph, this corresponds to costs at nodes and costs along the edges. There is one node in the graph for each phoneme instantiation in the unit database, and the transition costs along the edges between two nodes represent concatenation of the nodes. Although we shall discuss transition costs later, we first note that two phonemes that are spoken in succession should have no transition cost. Also, we observe that these transition costs can be pre-computed to improve run-time performance.

5.4 Unit cost

As units are considered for selection, they must each be ranked as to how they compare with what is requested. We can quantify this distance by defining a unit cost function that compares how close two units are. We consider the phonemes in the input specification to be the "truth." For a given phoneme, we use the unit cost function to obtain a ranked list of the multiple instantiations of the phoneme within the speech database. Any deviation from the truth is calculated and tabulated to be the unit cost. The "closest" phoneme is not necessarily always selected, because there also is a transition cost that describes the cost incurred in traveling from the current phoneme to the phoneme in question. The transition cost will be considered in the following section.

The unit cost function incorporates two types of distances: a co-articulatory distance and a prosodic distance. The co-articulatory distance is inspired by the desire to match the place of articulation at the left and right boundaries of the phoneme. The prosodic distance was not experimentally investigated in this work, but provided in the implementation for completeness. If prosody information is provided, a deviation along the duration, pitch, and energy dimensions is also calculated. This is accomplished by measuring the distance in seconds, Hz, and dB, respectively. We shall concentrate only on the co-articulatory distance.

The co-articulatory distance is measured by considering trigram context, namely the left and right neighbors of the phoneme to be scored. A cost is incurred if the phonemic context on the left and right sides do not match exactly; an exact phonemic context is zero cost. In the case the phonemic context does not match, some alternate contexts may be preferable over others. By defining context classes, we can allow for graceful back-off to other phonemic contexts within the same class that would give rise to similar co-articulatory behavior.

Instead of defining unit cost functions for each phoneme, we defined unit cost functions over manner classes. That these manner classes individually have consistent co-articulatory behavior justified such a decision. Specifically, we considered vowels/semivowels, stops, nasals, silence, and a final group that included fricatives, affricates, and the glottal phoneme, h. This is displayed in Table 5-1. The last group was formed even though its three constituents can have dissimilar behavior, i.e. affricates can exhibit stop-like behavior as they can have closures. Fundamentally, however, the three types of sounds are all formed by generating noise (frication or aspiration) through a constriction in the vocal tract.

vowel:	y iy ih ix ux ey eh ae ay oy uw uh ah ax ow ox er rx aa ao aw w l r el
stop:	b d g p t k
nasal:	m em n en nx
fricative:	f v s z th dh dx sh zh ch jh hh
silence:	h# pau

Table 5-1: Manner classes for unit cost function.

Because the co-articulatory characteristics of what may lie to the left and right can vary across manner classes, a separate set of left and right context classes was defined for each manner class. Finally, each manner class has a cost matrix for the left and right side indexed by the members of the left and right context classes. Generally, the cost matrices possess diagonal entries with low costs and off-diagonal entries with high costs; however, we will see some exceptions later. It should be noted that low costs must have a non-zero floor cost. Otherwise, there would be no distinction between an exact match of phonemic context or just a class match of phonemic context.

5.4.1 Vowel unit cost

For vowels, we defined the following identical sets of context classes for the left and right sides in Tables 5-2 and 5-3, respectively. These identical sets of context classes for the left and right sides of a vowel stem from the seven classes in Chapter 4. The place of articulation and nasalization are important factors, whereas the voicing dimension could be safely ignored to a first order approximation. The final two classes, *front* and *back*, were an attempt to ensure the formant continuity of *F2*, which we hypothesized to be important. Note that these were only used in vowel-vowel sequences which required a splice. Finally, diphthongs belong to either the *front* or *back* class depending on whether they occur on the left or right side.

labial:	b p f v w
dental:	d t s z sh zh ch jh th dh dx
velar:	g k
nasal_labial:	m em
nasal_dental:	n en
nasal_velar:	nx
front:	y iy ih ix ux ey eh ae ay oy
back:	uw uh ah ax ow ox er rx aa ao aw l r el
none:	hh h# pau

Table 5-2: Left context classes for vowel unit cost function.

labial:	b p f v
dental:	d t s z sh zh ch jh th dh dx
velar:	g k
nasal_labial:	m em
nasal_dental:	n en
nasal_velar:	nx
front:	y iy ih ix ux ey eh ae aw
back:	uw uh ah ax ow ox er rx aa ao ay oy w l r el
none:	hh h# pau

Table 5-3: Right context classes for vowel unit cost function.

With the context classes defined, we present the vowel unit cost matrix for the left side in Table 5-4. The rows represent the context class of the desired input specification and the columns represent the context class of a proposed unit from the unit database. Similarly, on the right side we have the vowel unit cost matrix in Table 5-5. The right cost matrix for vowels in Table 5-5 is almost similar to its left side counterpart except that it has some local variation representing speech knowledge about nasalization. When selecting a nasalized vowel (nasal on the right side) from the unit database, it is not entirely bad to choose a non-nasalized vowel. This is encoded by a moderate cost of 100. However, the converse is not true, and thus this local variation is not symmetric. From listening, we have observed that choosing a non-nasalized vowel can cause the nasal murmur of the following nasal to be perceived louder. This can be attributed to the abrupt lowering of the velum. When a vowel is nasalized, the velum articulator tends to lower ahead of the oral closure.

	labial	alveolar	velar	m	n	ng	front	back	none
labial	10	1000	1000	1000	1000	1000	1000	1000	1000
alveolar	1000	10	1000	1000	1000	1000	1000	1000	1000
velar	1000	1000	10	1000	1000	1000	1000	1000	1000
m	100	1000	1000	10	1000	1000	1000	1000	1000
n	1000	100	1000	1000	10	1000	1000	1000	1000
ng	1000	1000	100	1000	1000	10	1000	1000	1000
front	1000	1000	1000	1000	1000	1000	10	1000	1000
back	1000	1000	1000	1000	1000	1000	1000	10	1000
none	1000	1000	1000	1000	1000	1000	1000	1000	10

Table 5-4: Left cost matrix for vowel unit cost function.

	labial	alveolar	velar	m	n	ng	front	back	none
labial	10	1000	1000	1000	1000	1000	1000	1000	1000
alveolar	1000	10	1000	1000	1000	1000	1000	1000	1000
velar	1000	1000	10	1000	1000	1000	1000	1000	1000
m	100	1000	1000	10	1000	1000	1000	1000	1000
n	1000	100	1000	1000	10	1000	1000	1000	1000
ng	1000	1000	100	1000	1000	10	1000	1000	1000
front	1000	1000	1000	1000	1000	1000	10	1000	1000
back	1000	1000	1000	1000	1000	1000	1000	10	1000
none	1000	1000	1000	1000	1000	1000	1000	1000	10

Table 5-5: Right cost matrix for vowel unit cost function.

5.4.2 Fricative unit cost

Allophonic variations of stops can often be attributed to round and retroflexed environments to the left and right. If these variations are not explicitly accounted for, unnatural sounding speech can result from the cross-usage of these variations.

For fricatives, we defined the following sets of context classes for the left and right sides in Tables 5-6 and 5-7, respectively. As before in the vowel context classes, dipthongs change classes depending on whether they are on the left or right side of a fricative. An even more important distinction between the left and right sides of a fricative is seen in the placement of labial consonants into the *round* class. This accounts for allophonic variations such as a labial tail in the cut-off frequency of a fricative. Both labial consonants and round sonorants effectively lengthen the vocal tract, which tends to lower the corresponding resonance frequencies.

`retroflex:`	`r er rx`
`round:`	`uw ux w ow ox aw`
`sonorant:`	`y iy ih ix ey eh ae ay oy uh ah ax aa ao l el m em n en nx`
`other:`	`b p f v d t s z sh zh ch jh th dh dx g k hh h# pau`

Table 5-6: Left context classes for fricative unit cost function.

`retroflex:`	`r er rx`
`round:`	`uw ux w b p f v m em`
`sonorant:`	`y iy ih ix ey eh ae ay oy aw uh ah ax ow ox aa ao l el n en nx`
`other:`	`d t s z sh zh ch jh th dh dx g k hh h# pau`

Table 5-7: Right context classes for fricative unit cost function.

For the left and right side of fricatives, we have identical cost matrices in Tables 5-8 and 5-9, respectively.

retroflex	10	100	100	100
round	100	10	100	100
sonorant	100	100	10	100
other	100	100	100	10

Table 5-8: Left cost matrix for fricative unit cost function.

retroflex	10	100	100	100
round	100	10	100	100
sonorant	100	100	10	100
other	100	100	100	10

Table 5-9: Right cost matrix for fricative unit cost function.

5.4.3 Stop unit cost

Allophonic variations of stop consonants can often be attributed to two factors: flapping, and the presence of front, back, round, or retroflexed environments to the left and right [33]. If these variations are not explicitly accounted for, unnatural sounding speech can result from the cross-usage of these variations.

For stops, we defined the following sets of context classes for the left and right sides in Tables 5-10 and 5-11, respectively.

```
front:     y iy ih ix ey eh ae ay oy
back:      uh ah ax aa ao l el
retroflex: r er rx
round:     uw ux w ow ox aw
other:     b p f v d t s z sh zh ch jh th dh dx g k m em n en nx
  hh h# pau
```

Table 5-10: Left context classes for stop unit cost function.

For the left and right side of stops, we have the following cost matrices in Tables 5-12 and 5-13, respectively.

The cost matrices for stops have slightly different structure from the matrices before. Because stops generally have a closure on the left side, the co-articulation through this silence region is not totally important. However, when a consonant precedes the stop - as in a consonant cluster in the onset position - it is important to match the context. For example, unvoiced stops are not aspirated when in a cluster with an /s/ (e.g., ski, spot.) We will also treat this phenomenon later in the transition cost.

On the right side of a stop, we take special measures to ensure that schwas are allowed at a low cost into the unit only when requested, and vice versa. An alveolar stop with a unstressed syllable to the right, such as a schwa, may be realized as a flap. These costs guarantee protection against cross-usage between fully released stops and flapped stops. Note that an alternative procedure would have been to predict flapping from the baseform, and use a different phonetic symbol (ɾ) in the search.

```
front:      y iy ih ix ey eh ae aw
back:       uh ah ow ox aa ao ay oy l el
retroflex: r er rx
round:      uw ux w
schwa:      ax
other:      b p f v d t s z sh zh ch jh th dh dx g k m em n en nx
  hh h# pau
```

Table 5-11: Right context classes for stop unit cost function.

	front	back	retroflex	round	other
front	10	10	10	10	10
back	10	10	10	10	10
retroflex	10	10	10	10	10
round	10	10	10	10	10
other	500	500	500	500	10

Table 5-12: Left cost matrix for stop unit cost function.

	front	back	retroflex	round	schwa	other
front	10	100	100	100	500	100
back	100	10	100	100	500	100
retroflex	100	100	10	100	500	100
round	100	100	100	10	500	100
schwa	500	500	500	500	10	500
other	100	100	100	100	500	10

Table 5-13: Right cost matrix for stop unit cost function.

5.4.4 Nasal unit cost

Allophonic variations of nasal consonants can be mainly attributed to syllable position (onset or coda) and durational lengthening. Regarding lengthening, voiced stops to the right of a nasal tend to give the nasal a longer duration [12]. Cross-usage of these variations can even confuse the listener as to whether the following stop is voiced or not. For example, if a lengthened nasal were to be concatenated with an unvoiced stop, a listener might perceive a voiced stop anyways. Therefore, the dimension of voicing is important on the right side of a nasal.

For nasal, we defined the following sets of context classes for the left and right sides in Tables 5-14 and 5-15, respectively.

```
obstruent: b p f v d t s z sh zh ch jh th dh dx g k hh h# pau
sonorant:  y iy ih ix ux ey eh ae aw uw uh ah ax ow ox er rx aa ao
  ay oy w l r el m em n en nx
```

Table 5-14: Left context classes for nasal unit cost function.

```
voiced:   b v d z zh jh dh g dx h# pau
unvoiced: p f t s sh ch th k hh
sonorant: y iy ih ix ux ey eh ae aw uw uh ah ax ow ox er rx aa ao
  ay oy w l r el m em n en nx
```

Table 5-15: Right context classes for nasal unit cost function.

To distinguish between the excitation source, the context classes on the left side are merely divided into the obstruents and the sonorants. On the right side of a nasal, we must further divide the obstruent class into voiced and unvoiced classes to account for the durational lengthening phenomenon in speech production. As pre-pausal speech tends to be lengthened as well, we include silence as an approximation into the voiced class to further benefit from this modeling of lengthening.

For the left and right side of nasals, we have the following cost matrices in Tables 5-16 and 5-17, respectively.

	obstruent	sonorant
obstruent	10	1000
sonorant	1000	10

Table 5-16: Left cost matrix for nasal unit cost function.

	voiced	unvoiced	sonorant
voiced	10	100	1000
unvoiced	100	10	1000
sonorant	1000	1000	10

Table 5-17: Right cost matrix for nasal unit cost function.

While the vowel unit cost will cause a nasal consonant to be selected from the correct syllable position (onset or coda), the nasal unit cost function serves to select an inter-vocalic nasal in an inter-vocalic position (sonorant-nasal-sonorant), a pre- or post-vocalic nasal when required, and to deal with lengthening.

The right cost matrix accounts for durational lengthening effects. However, it is somewhat forgiving of cross-usage as shown by the weight entries of 100 in a voiced-unvoiced confusion, as long as the nasal is in the same syllable position.

5.5 Transition cost

The unit cost function described above quantifies how two units compare. It is used to weigh how a unit would fare if selected in place of the "truth." As mentioned before, the "closest" unit is not necessarily selected, because there remains the issue of how the units connect. If spoken in succession, two phonemes have zero transition cost. But, when they originate from separate places, there can be varying degrees of acceptability of the join. Two successive units with sub-optimal co-articulatory context may be preferable over two non-adjacent units with optimal co-articulatory context. Therefore, it is the job of the transition cost function to make the trade-offs when joining units that are optimal in their own right.

The transition cost function incorporates two types of continuity measures: a co-articulatory continuity measure and a prosodic continuity measure. The first is inspired by the fact that certain phonemes spoken in succession exhibit a significant amount of co-articulation, or formant motion; phoneme pairs as such should certainly not be compromised by a concatenation that may bring together two non-contiguous phoneme instantiations whose formants would not smoothly connect. Prosodic continuity measures were not explicitly experimentally investigated in this work, nor provided in the implementation, besides our treatment of lexical stress to be described later. Past work by others have considered spectral, formant, pitch, and energy continuity measures [17]. We focus on the co-articulatory continuity measures.

When two phonemes are proposed for concatenation, if they were not spoken in succession, a transition cost must be incurred. The transition cost function then quantifies and ranks what type of phoneme pairs can be created by concatenation of two non-contiguous phoneme constituents. Sometimes the importance of co-articulatory continuity can override the importance of the individual unit costs of the two phonemes. As before in the unit cost function, we again allow for graceful back-off to manner pairs that would give rise to co-articulatory continuity across the concatenation boundary.

Defining transition cost functions over manner-manner pairs not only reduces the size of the function space, but also allows for modeling of higher-level constraints. The manner classes that we define are fairly consistent with standard English phoneme taxonomy. We consider six classes: vowels, semivowels, nasals, /h/, obstruents, and silence. This is displayed in Figure 5-18.

vowel:	y iy ih ix ux ey eh ae ay oy uw uh ah ax ow ox er rx aa ao aw
semivowel:	w l r el
nasal:	m em n en nx
glottal:	hh
obstruent:	b p f v d t s z sh zh ch jh th dh dx g k
silence:	h# pau

Table 5-18: Manner classes for transition cost function.

The reason for /h/ occupying a class by itself arises from perceptual experiments showing that it adapts to its co-articulatory environment. /h/ is produced by aspiration passing through a constriction after the glottis. The output can be highly variable as the output takes on the characteristics of the current configuration of the vocal tract. In an inter-vocalic position, /h/ tends to adopt the resonance frequencies of the neighboring vowels.

5.5.1 Modeling sub-syllabic structure

The definition of these manner classes places us in a position to model higher-level constraints. Figure 5-1 depicts sub-syllabic structure from classic syllable theory [24]. As circles mark optional paths, a minimal syllable consists of just a nucleus. An affix is generally only realized in a pre-pausal syllable. Affixes are composed by word-final consonant clusters.

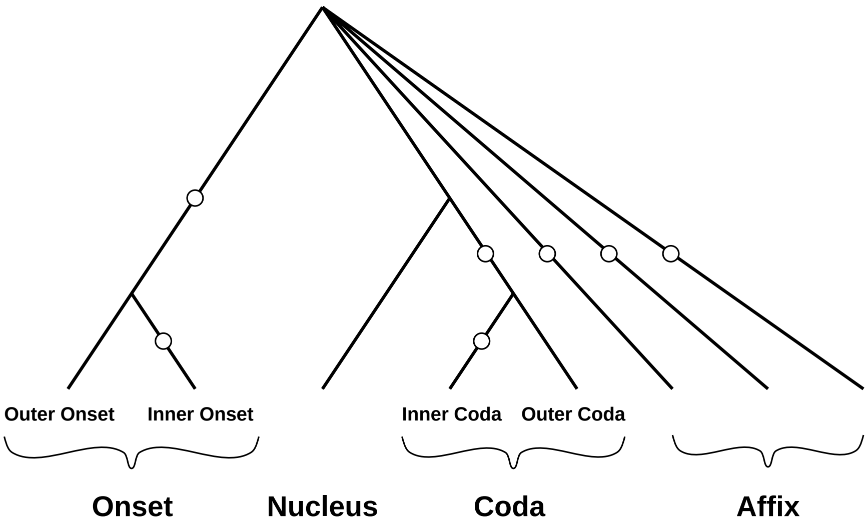


Figure 5-1: Sub-syllabic structure [24].

To understand how the transition cost can be used to model sub-syllabic structure relations, we must understand how the transition manner classes correlate with sub-syllabic structure. Table 5-19 shows where certain sound classes can exist within a syllable. Note that silence is left out, because it is not part of a syllable, automatically implying a syllable boundary.

	Onset	Nucleus	Coda	Affix
Vowel		X		
Semivowel	X		X	
Nasal	X		X	
/h/	X			
Obstruent	X		X	X

Table 5-19: Where sound classes can be realized within syllable.

As Table 5-19 shows, within a syllable, a vowel preceded by a semivowel, nasal, glottal, or obstruent implies an onset-nucleus pair. The reverse argument leads to a nucleus-coda pair. Across a syllable boundary, a vowel preceded by a semivowel, nasal, or obstruent is clearly a coda-nucleus pair, implying the absence of an onset in the second syllable. The reverse argument leads to a nucleus-onset pair, implying the absence of a coda in the second syllable. We will explore more relationships later on. However, this preliminary analysis shows that there is much to be gained in terms of sub-syllabic modeling power from the transition cost function.

Two separate cost matrices are maintained for transitions occurring within or across syllables. We term these costs the intra-syllable transition and inter-syllable transition cost functions, respectively. Naturally, the application of the above relationships requires knowledge of where syllable boundaries lie. This decoupling allows the separate investigation of what manner-manner pairs should be protected from concatenation within and across syllables.

5.5.2 Intra-syllable transition cost

The cost matrix for non-adjacency of two units within a syllable is depicted in Table 5-20. The rows represent the left side of the transition and the columns represent the right side of the transition. *D* stands for "don't-care" costs.

	vowel	semivowel	nasal	obstruent	/h/
vowel	10000	10000	7500	10	D
semivowel	10000	7500	7500	10	D
nasal	5000	10	D	10	D
/h/	5000	D	D	D	D
obstruent	10	10	10	10000	D

Table 5-20: Intra-syllable cost matrix for transition cost function.

These non-adjacency costs implicitly encode many types of knowledge including speech production and sub-syllabic structure. First, the only time a glottal can appear within a syllable is in the onset position with a vowel nucleus. Every other entry involving /h/ is marked with *D*. The nasal-nasal entry also is marked with *D* for obvious reasons. Second, since we have designed our sub-word unit corpus with VV, SV, and VS sonorant sequences in mind, we associate high costs with their non-adjacency. Next, a pair involving an obstruent and any sonorant sound implies a source change. The associated non-adjacency cost is low, and so concatenations can be forgiven. Finally, to capture allophonic variations of obstruents in clusters, the cost with breaking an obstruent-obstruent sequence is high. For example, this can help to obtain contiguously unvoiced stops in a cluster with an /s/ (e.g., ski, spot).

5.5.3 Inter-syllable transition cost

The cost matrix for non-adjacency of two units across a syllable boundary is depicted in Table 5-21.

	vowel	semivowel	nasal	obstruent	/h/	silence
vowel	D	7500	5000	10	5000	10
semivowel	7500	7500	2000	10	10	10
nasal	2000	10	10	10	10	10
obstruent	10	10	10	5000	10	10
/h/	D	D	D	D	D	D
silence	10	10	10	10	10	10

Table 5-21: Inter-syllable cost matrix for transition cost function.

Again, we see how these non-adjacency costs implicitly encode many types of knowledge. As glottals can never appear on the left side of a syllable boundary, the entire glottal row is marked with *D*'s. It is important to mention that the vowel-glottal is relatively high, because such a sequence implies an intervocalic glottal; the glottal must be followed by another vowel. This will encourage the selection of a contiguous vowel-glottal-vowel sequence which has strong formant dynamics. As our syllabification algorithm places all contiguous vowels into the nucleus, a vowel-vowel sequence will never occur across syllables, and is marked with *D*. For the VV, SV, and VS sequences, we see high non-adjacency costs. The relatively high obstruent-obstruent cost helps to capture allophonic variations. As a final note, the inter-syllable transition costs are generally lower than intra-syllable transition costs, because contiguity preservation is more important within a syllable than across syllables.

5.6 Combining costs

With the unit and transition cost functions both possessing a component inspired by speech production and a potential prosodic component, we need to combine these components in a meaningful manner to obtain a sensical unit and transition cost.

Because prosodic distances and continuity measures were not pursued in this work, the unit and transition cost functions were simply determined by the co-articulatory distance and continuity measures alone.

To allow the transition cost function to override the unit cost function when necessary, we compute the Viterbi cost at a node in a given time slice as follows: the new cost at the arriving node is the equally-weighted sum of the cost at the departing node in the previous time slice, the transition cost between the two nodes, and the unit cost of the arriving node. If the Viterbi search wishes to explore a path to a optimal unit, it must optimize the transition cost along the way as well. For example, the intra-syllable transition cost function seeks to preserve the continuity of vowel and semivowel sequences.

5.7 Pragmatic considerations

The units designed in Chapter 4 have no concept of stress; stress markers are removed from the pronunciations in the PRONLEX lexicon. As a result, the phones in the Viterbi lattice are also not marked with stress information. However, as stress can play a role in determining duration, the phone graph was at times insufficient as cross-usage of phonemes varying differently in duration became apparent, for example, in poly-syllabic words. To combat this, unstressed versions of certain phonemes were treated as a special reduced unit and given reduced labels:

Unstressed	Reduced
o^{w}	ox
ɪ	ix
ɝ	rx
u^{w}	ux

The reason these units were identified as needing separate reduced labels stems from the fact that these four phonemes occupy disjoint regions in a vowel space plot where the axes are *F1* and *F2*. All four phonemes possess distinct formant structures, and two are also dipthongs with offglides producing formant motion.

These four mappings helped to alleviate cases when the prosody did not sound natural in poly-syllabic words. However, discarding lexical stress from PRONLEX pronunciations removes information in a irreversible manner. Improving the naturalness of poly-syllabic synthesis is left as a task for future research.

5.8 N-best synthesis

If we wish to obtain an N-best list of paths through the graph, we can take the resulting Viterbi costs at each node and turn them around for use as underestimates in an A* search [27]. This A* search is performed backwards through the lattice and performs well if the Viterbi cost is a good measure.

Since the A* search is a queue-based best-first search, we must consider the pragmatic issue of path explosion. Because the backwards A* search is performed over the Viterbi lattice, dynamic programming pruning can be performed at each time slice. However, if many nodes exist in each time slice, the branching factor can be large, thereby overflowing the path queue. This is handled by an empirical A* pruning threshold, where paths are only extended to nodes if they are within a threshold of the best node in their time slice.

5.9 Discussion

This chapter has delineated the search algorithm used for sub-word unit selection. Operating over a phoneme graph, a Viterbi search optimizes a search metric based on individual and ensemble considerations of sub-word unit sequence candidates. These correspond to the unit cost and transition cost functions, respectively.

The unit cost function ranks units on an individual basis against the "truth." It pursues the matching of formant loci at the left and right unit boundaries by matching phonemic contexts on the left and right sides. When necessary, it will back-off to manner classes possessing similar formant loci as learned from previous perceptual experiments.

The transition cost function rates unit sequence candidates for overall continuity. As the transition cost functions seeks to maximize the size and contiguity of speech segments, thereby minimizing the number of concatenations, it encourages the selection of multi-phoneme sequences, or sub-word units. Simultaneously, it also dictates when concatenations can occur with minimal perceptual distortions and when phoneme-phoneme pairs must be protected from splicing due to strong co-articulatory and formant motion. Where preservation from concatenation should be enforced can be learned from the examination of sub-syllabic structure. Although the transition cost function considers whether concatenation should be allowed within, or across syllables, it can not guarantee if a unit destined for usage in an onset position was actually realized in the onset position, for example. Utilizing syllable boundary information within the unit cost function should be examined in future work.

We have seen how the unit cost and transition cost functions can deal with speech production phenomena. For example, the unit cost function can implicitly account for lengthening and nasalization. The transition cost function can implicitly account for the falling of the cut-off frequency in fricatives followed by a phoneme produced by labial constriction and how unvoiced stops are unaspirated in consonant clusters.

When the unit cost and transition cost functions are combined to form the Viterbi search metric, there can be interaction amongst the two. For example, within a syllable, the transition cost function can override the unit cost function to preserve the continuity of vowel and semivowel sequences.

The cost numbers presented in the matrices of this chapter represent hand-adjusted weights. As the context class and cost matrix definitions reside in user-editable files, they can be tuned at any time should other speech phenomena previously unaccounted for produce unnatural speech synthesis.

The compact corpus designed in Chapter 4 covered a set enumerating all sub-word units required for the synthesis of any word from the non-foreign words of the PRONLEX lexicon. But the search graph nodes presented in this chapter were not sub-word units, but rather phonemes; the sub-word unit labels were discarded. By relaxing the minimal unit down to the phoneme level, the search algorithm could on its own find the optimal temporal granularity of the units at search time. This can be thought of as a “soft” assignment, as opposed to a “hard” assignment imposed by using the sub-word unit labels directly.

The solution to selecting sub-word units from a speech database pursued in this work is an indirect method. The Viterbi search optimizes a forced path through a graph of phonemes. Multi-phoneme sequences are extracted under the direction of the transition cost function. When desirable sub-word units are present within the speech database, the search algorithm will seek them out. This solution presents a late-binding technique. However, novel test data (i.e., foreign pronunciations) would never present the system with a “new unit” problem provided their pronunciations are mapped to English units. With complete coverage in the training data, this search algorithm will perform as well as one that directly searched on a graph of sub-word units. In the case of sparse training data, it will not give up, but rather back off and perform to its best ability with the data available.

This search framework has demonstrated the generic ability to select units from any speech corpus that has been phonemically aligned. However, there seem to be limitations regarding the range of the distance metric and lexical stress in poly-syllabic words. In an /str/ consonant cluster, the cut-off frequency in the /s/ can be affected by the retroflexion in the phoneme two positions away. Because the transition cost function tends to maximize sequence contiguity, it is possible to select this cluster. A more direct way to specify the selection of this cluster would be to allow for longer-distance measures in the unit cost function (e.g., the /s/ is in an /str/ cluster.) But, this can also have the disadvantage of a larger unit space. In regard to lexical stress, we have empirically seen the inadequacy of discarding stress markers. This resulted in cross-usage of sub-word units in poly-syllabic words that did not produce an natural-sounding prosody. This may be attributable to the initial design of the unit inventory Chapter 4, or not explicitly including prosodic measures in the unit cost function. The problem of dealing with lexical stress is left to be treated in future work.

Chapter 6

Corpus-based sub-word concatenation examples

6.1 Introduction

In Chapter 3, we described perceptual experiments in which we explored unit and transition costs that determine how units score individually and as an ensemble, respectively. This helped us to enumerate units in the English language and prepare a compact corpus with full coverage in Chapter 4. Next, in Chapter 5, we devised a search algorithm to integrate the unit and transition criterions learned from the perceptual experiments to select an optimal unit sequence from the designed corpus. Now, we are prepared to combine these components to form a complete sub-word concatenative synthesis system.

In this chapter, we shall examine the types of decisions the search algorithm makes as it select units from the unit database. We shall see how costs can be tuned to achieve the types of results desired.

6.2 Architecture

A general diagram illustrating the execution flow of a sub-word concatenative synthesizer is depicted in Figure 6-1. The context class and cost matrix files represent the unit constraints block. A sub-word corpus comprises the unit database. A pronunciation is generated from an input sequence of words. This pronunciation along with an overall sentence prosody is used to generate a prosody. The pronunciation and prosody are used by the unit and transition cost functions in the sub-word unit search. Finally, the speech waveform can be prosodically modified if desired.

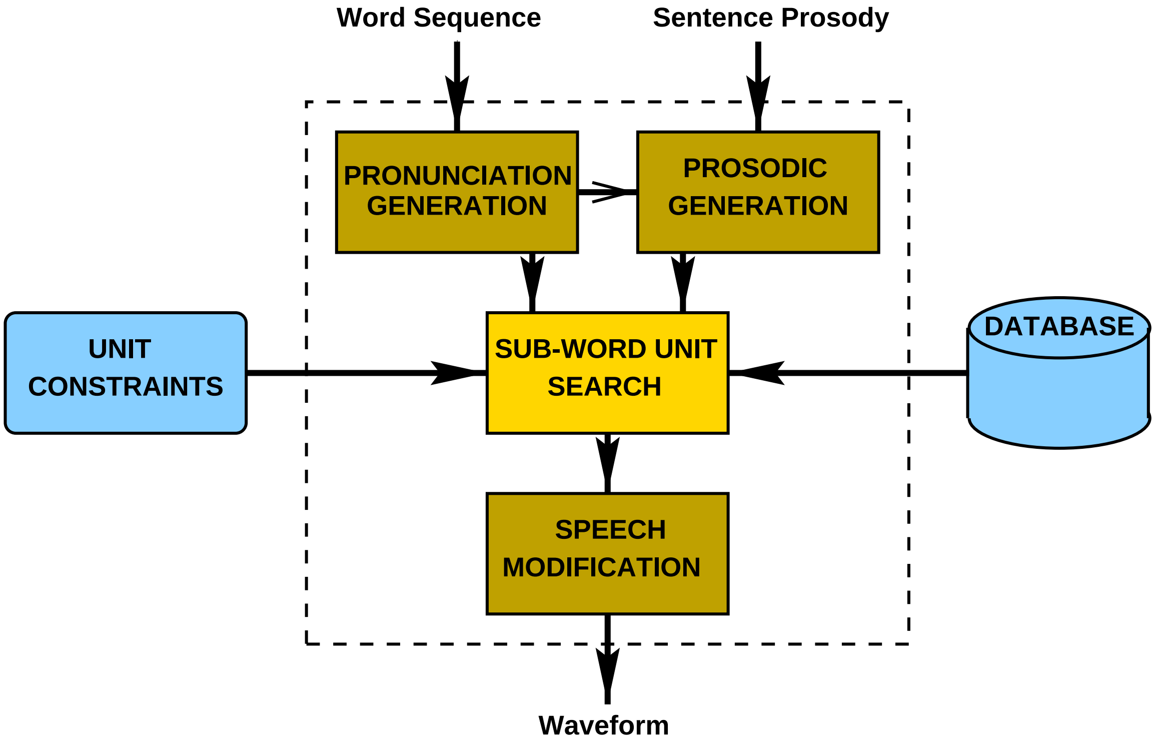


Figure 6-1: Sub-word concatenative synthesis architecture.

Within the synthesis engine, we focused our research on the search component. As for the pronunciation, prosodic, and modification components, we have either no, or pathological implementations. For example, the pronunciation generation component involves a lookup from a static lexicon, such as the 90,000-word PRONLEX, of which we only utilized a 68,000-word non-foreign subset with lexical stress markers discarded. The components from the previous three chapters come together in the middle row of the architecture diagram. Although not generally used in this work, we have explored using TD-PSOLA for speech modification [32].

6.3 Speech production

This section explores several different speech production phenomena, and how they are treated by the sub-word search algorithm. Onset and coda stop consonant clusters, duration lengthening, and labial trajectories in fricative cut-off frequencies are treated.

6.3.1 Onset stop consonant cluster

In this subsection, we examine the minimal pair of "pace" and "space" to demonstrate how the transition cost function is correctly capturing an unaspirated variation of unvoiced stops when in a onset cluster with /s/. In Figure 6-2, we see the monosyllabic word, *pace* /p e^{y} s/, synthesized using sub-word units from three words: *panorama*, *incubator*, and *clays*. Let's examine the mechanics of the underlying search decisions:

The /p/ selected from *panorama* is not optimal, because the right side of what was requested was an /e^{y}/, whereas what was on the right side of what was selected was an /æ/. However, since both are front vowels on the right side of a stop, the co-articulatory match is adequate. Because we are leaving the /p/ in *panorama* and transitioning into the /e^{y}/ from *incubator*, we incur a slight transition cost. The cost is only slight, because within a syllable, obstruent-vowel transitions are fairly low cost.

The unit cost of the /e^y/ selected from *incubator* is not bad. At least, labial /b/ and alveolar /t/ match the place of articulation of /p/ and /s/ (the /t/ also matches voicing) respectively, so the formant motion of the vowel should be similar.

Since part of the vowel following an unvoiced stop will be devoiced, and the vowel boundaries used in this work were based on voicing, there is likely too much formant motion in sequences of unvoiced stops followed by vowels taken from voiced stops and too little formant motion in sequences of voiced stops followed by vowels taken from unvoiced stops. However, this phenomenon was not perceived to be a major effect in practice. Future work would explore using a subset of voiced stop to vowel sequences, or including aspiration as part of the vowel.

Next, as we transition from the /e^y/ in *incubator* to the /s/ in *clays*, we again incur a slight transition cost, yet the cost is again slight, because within a syllable, vowel-obstruent transitions are fairly low cost. Finally, the /s/ phoneme obtained is optimal, because /e^y/ and silence are present on the left and right sides, respectively. Note that *clays* phonemically requires a /z/, but in this instance was transcribed as an /s/ as it was devoiced, and was selected because the search engine operates at the phonetic level.

In Figure 6-2, we also see the mono-syllabic word, *space* (/s p e^y s/), synthesized using sub-word units from three constituent words. Let's examine the mechanics of the underlying search decisions:

The /sp/ unit selected from *spookiest* is not optimal, because the right side of what was requested was an /e^{y}/, whereas what was on the right side of what was selected was an /u^{w}/. Thus, co-articulatory match is less optimal than the match above, because this time there is a class conflict on the right side between a front and a back/round vowel. This is the price to pay when the unit database is not complete. By having a high intra-syllable obstruent-obstruent transition cost, the search engine tries to find the consonant cluster as a unit. In this case, the /sp/ sequence preserves the effect that unvoiced stops are not aspirated when in a cluster with /s/. A secondary effect is the labial co-articulation in the /s/. Note that an onset /sp/ cluster was obtained rather than an offset cluster by virtue of the contextual constraints of silence (implicitly a word boundary) and vowel on the left and right sides, respectively.

Because we are leaving the /sp/ in *spookiest* and transitioning into the /e^{y}/ from *incubator*, we incur a slight transition cost. The cost is only slight, because within a syllable, obstruent-vowel transitions are fairly low cost. The rest of the synthesis decisions are exactly as the case for *pace*.

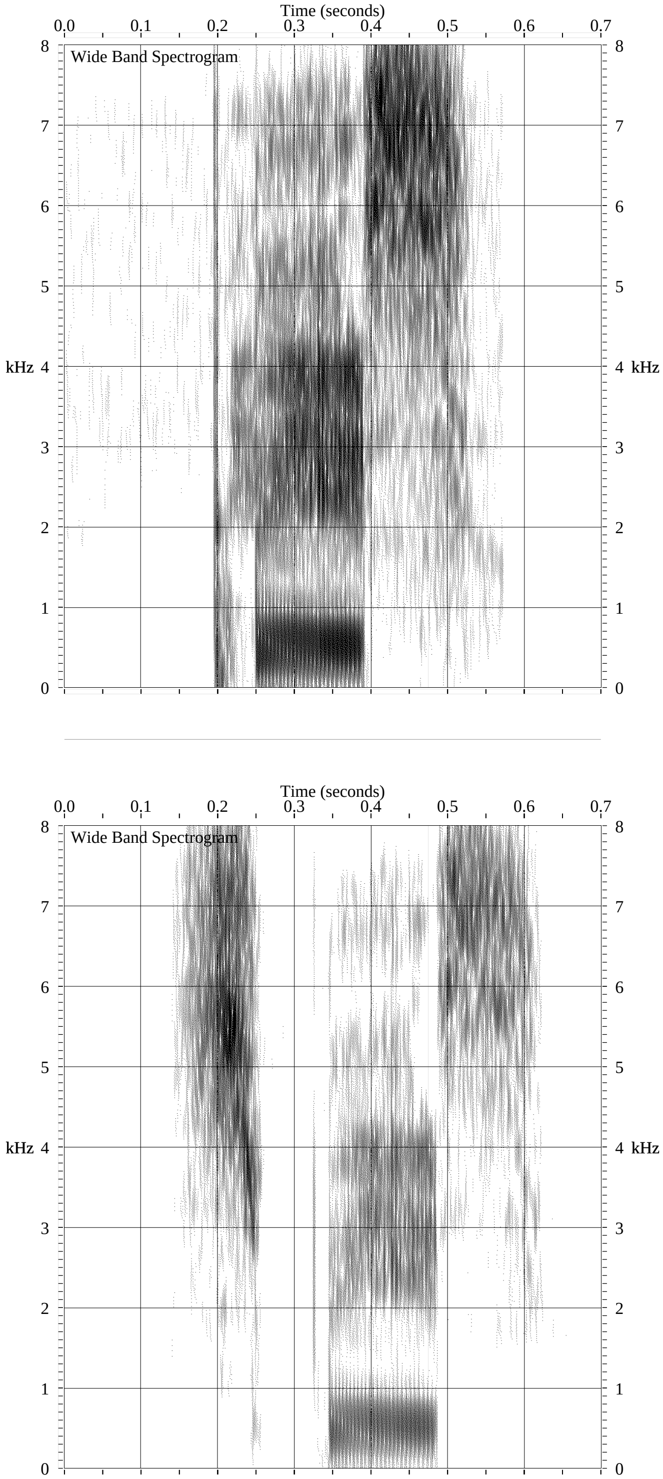


Figure 6-2: Spectrograms of synthetic “pace” and “space.”
“pace”: synthesized from [p]anorama incub[a]tor clay[s].
“space”: synthesized from [sp]ookiest incub[a]tor clay[s].
An unaspirated /p/ was correctly selected.

In these two synthesis examples, *pace* and *space*, the correct realization of the stop is selected, and the resulting waveforms sound natural. In *space*, an unaspirated /p/ was correctly selected, because of a high intra-syllable obstruent-obstruent transition cost. Without a high cost to encourage contiguity of two obstruents within a syllable, an aspirated version may have been selected, producing synthesis that sounds less natural, such as displayed in Figure 6-3.

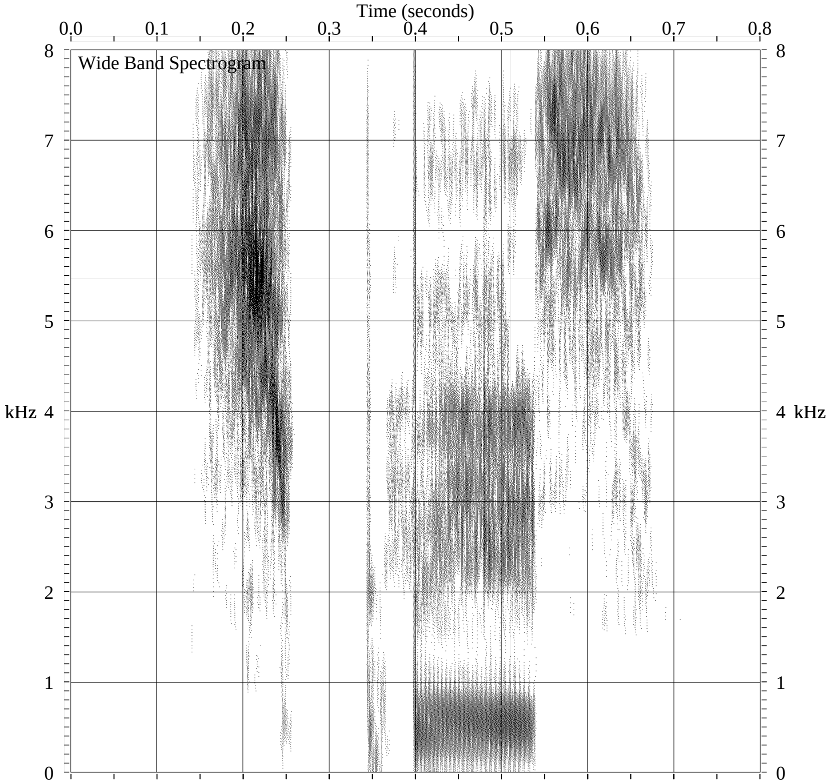


Figure 6-3: “space”: synthesized from [s]pookiest [p]anorama incub[a]tor clay[s]. Aspirated /p/ not natural in this context.

6.3.2 Coda stop consonant cluster

In this subsection, we examine the minimal pair of “sick and “six” to demonstrate how the transition cost function is correctly capturing an unaspirated variation of unvoiced stops when in a coda cluster with /s/. In Figure 6-4, we see the mono-syllabic word, *sick* /s ɪ k/, synthesized using sub-word units from three words: *syllabaries*, *nickelodeon*, and *candlewick*. Let’s examine the mechanics of the underlying search decisions:

The /s/ selected from *syllabaries* was optimal, because the left side (silence) and right side /ɪ/ were correctly matched. Because we are leaving the /s/ in *syllabaries* and transitioning into the /ɪ/ from *nickelodeon*, we incur a slight transition cost. The cost is only slight, because within a syllable, obstruent-vowel transitions are fairly low cost. The unit cost of the /ɪ/ selected from *nickelodeon* is not bad. At least on the left side, an alveolar /n/ is obtained in a class match. However, this may introduce some slight nasalization of the vowel as can possibly be seen in the spectrogram in the first formant region. On the right side, /k/ is correctly matched. As we transition from the /ɪ/ in *nickelodeon* to the /k/ in *candlewick*, we again incur a slight cost, yet the cost is again slight, because within a syllable, vowel-obstruent transitions are fairly low cost. Finally, the /k/ phoneme obtained is optimal, because /ɪ/ and silence (implicitly a word boundary) are present on the left and right sides, respectively. Also, it is correctly released and aspirated as it is post-vocalic and word-final.

In Figure 6-4, we also see the mono-syllabic word, *six* /s ɪ k s/, synthesized using sub-word units from the three words: *syllabaries*, *nickelodeon*, and *transfix*. Let’s examine the mechanics of the underlying search decisions:

Up to *transfix*, the synthesis decisions are exactly as before. Then, as we break from the /ɪ/ in *nickelodeon* to the /ks/ in *transfix*, we incur a slight transition cost, yet the cost is again slight, because within a syllable, vowel-obstruent transitions are fairly low cost.

The /ks/ unit obtained is optimal, because /ɪ/ and silence (implicitly a word boundary) are present on the left and right sides, respectively. Because of the high intra-syllable obstruent-obstruent transition cost, an unaspirated version of /k/ was correctly selected. Also, it was realized in a coda position by virtue of the contextual constraints of vowel and silence (implicitly a word boundary) on the left and right sides, respectively. This lends to the overall naturalness of the synthesized *six*.

In these two synthesis examples, *sick* and *six*, the correct realization of the stop is selected, and the resulting waveforms sound natural. In *six*, a /k/ which is unaspirated was correctly selected, because of a high intra-syllable obstruent-obstruent transition cost. Without a high cost to encourage contiguity of two obstruents within a syllable, an aspirated version may have been selected, producing synthesis that does not sound natural.

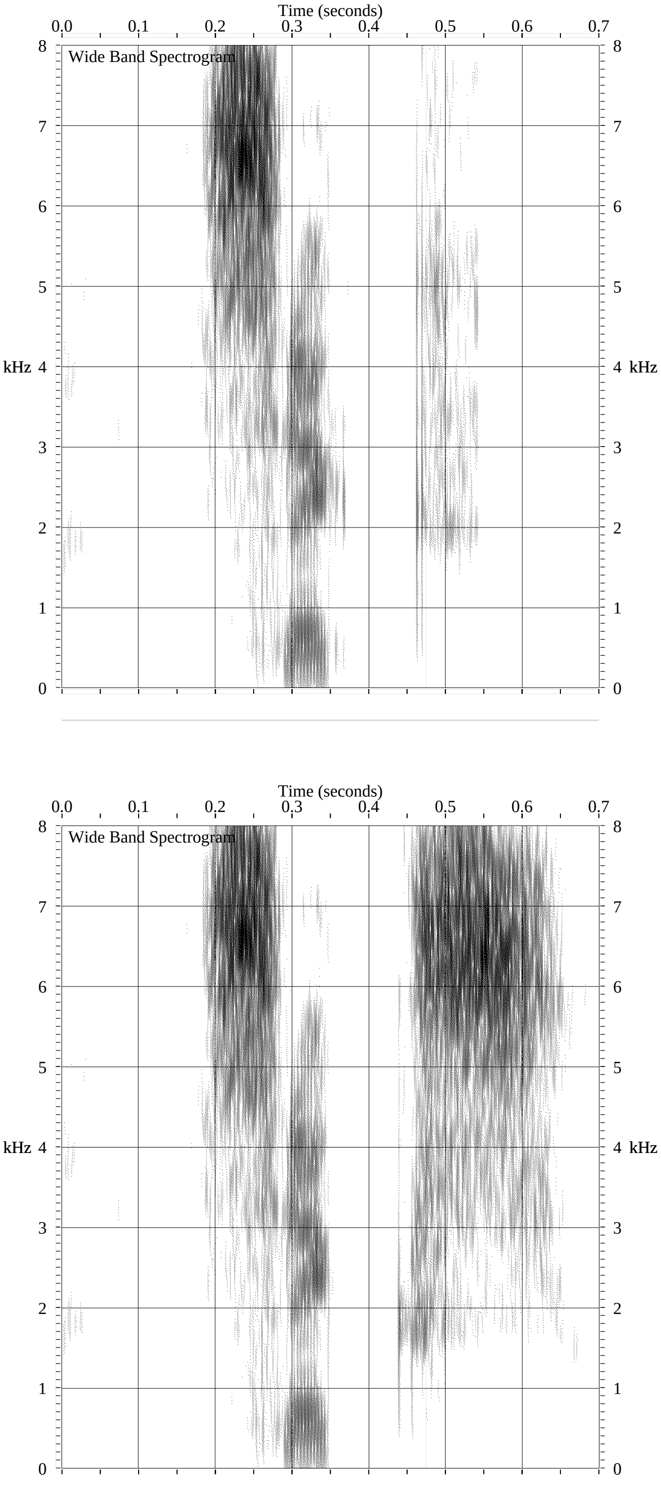


Figure 6-4: Spectrograms of synthetic “sick and “six.”
“sick”: synthesized from [s]yllabaries n[i]ckelodeon candlewi[ck].
“six”: synthesized from [s]yllabaries n[i]ckelodeon transfi[x].

6.3.3 Labial co-articulation in fricatives

In this subsection, we examine the pair of "scope" and "smoke" to demonstrate how the transition cost function is correctly capturing an allophonic variation of /s/ when in a onset cluster with a labial consonant. In Figure 6-5, we see the mono-syllabic word, *scope* /s k o^{w} p/, synthesized using sub-word units from the two words: *skylights* and *nope*. Let's examine the mechanics of the underlying search decisions:

The /sk/ unit selected from *skylights* is not optimal, because, although the left side (silence) was reasonably matched, the right side (/e^{y}/ versus /o^{w}/) was only a class (back) match. However, a cluster /sk/ in an onset position was correctly selected. Because we are leaving the /sk/ in *skylights* to select the /o^{w}p/ from *nope*, we incur a slight transition cost. The cost is only slight, because within a syllable, obstruent-vowel transitions are fairly low cost. Finally, the unit cost of the /o^{w}p/ unit selected from *nope* is not optimal. On the left side, alveolar /n/ is mismatched with the desired velar /k/ context. Also, the /o^{w}/ will be slightly nasalized. However, acquiring the /o^{w}/ and /p/ in succession and acquiring an aspirated version of the stop are redeeming qualities.

In Figure 6-5, we also see the mono-syllabic word, *smoke* /s m o^{w} k/, synthesized using sub-word units from three words: *swastika*, *nonsmoking*, and *dactylic*. Let's examine the mechanics of the underlying search decisions:

The /s/ selected from *swastika* was less than optimal, because, although the left side (silence) was correctly matched, the right side (/w/ versus /m/) was only a class (round) match. Fortunately, rounding the lips has the same effect on formant frequency locations as does labial constriction: it effectively increases vocal tract length, causing a drop in resonant frequencies. Hence, the cut-off frequency of the /s/ fricative has labial co-articulation. Because we are leaving the /s/ in *swastika* to select the /mo^{w}/ from *nonsmoking*, we incur a slight transition cost. The cost is only slight, because within a syllable, obstruent-nasal transitions are fairly low cost.

Selecting /mo^{w}/ from *nonsmoking* is optimal. The /s/ and /k/ phoneme contexts are matched on the left and right side, respectively. Next, as we break from the /mo^{w}/ in *nonsmoking* to the /k/ in *dactylic*, we again incur a slight transition cost, yet the cost is again slight, because within a syllable, vowel-obstruent transitions are fairly low cost.

Finally, the unit cost of the /k/ unit selected from *nope* is not optimal. On the left side, front /ɪ/ is mismatched with the desired round /o^{w}/ context, but on the right side, silence is correctly acquired. Note that while *smoke* could have been completely synthesized from *nonsmoking*, the /k/ in *nonsmoking* may have been too aspirated to use in an isolated version of *smoke*. This search result was changed by the addition of silence constraints on the right side of /k/ and should be investigated in future work.

In these two synthesis examples, *scope* and *smoke*, the correct realization of the fricative, /s/, is selected, and the resulting waveforms sound natural. In *smoke*, a /s/ possessing labial co-articulation was correctly selected - even when a silence/s/m is not available in the unit database - because of placing /w/ and /m/ into a round class on the right side of the fricative unit cost. Without such a fricative unit cost on the right side, an /s/ phoneme possessing a labial tail could be selected when a non-labial place of articulation is desired, or vice versa, leading to speech synthesis that does not sound natural, or is confusing.

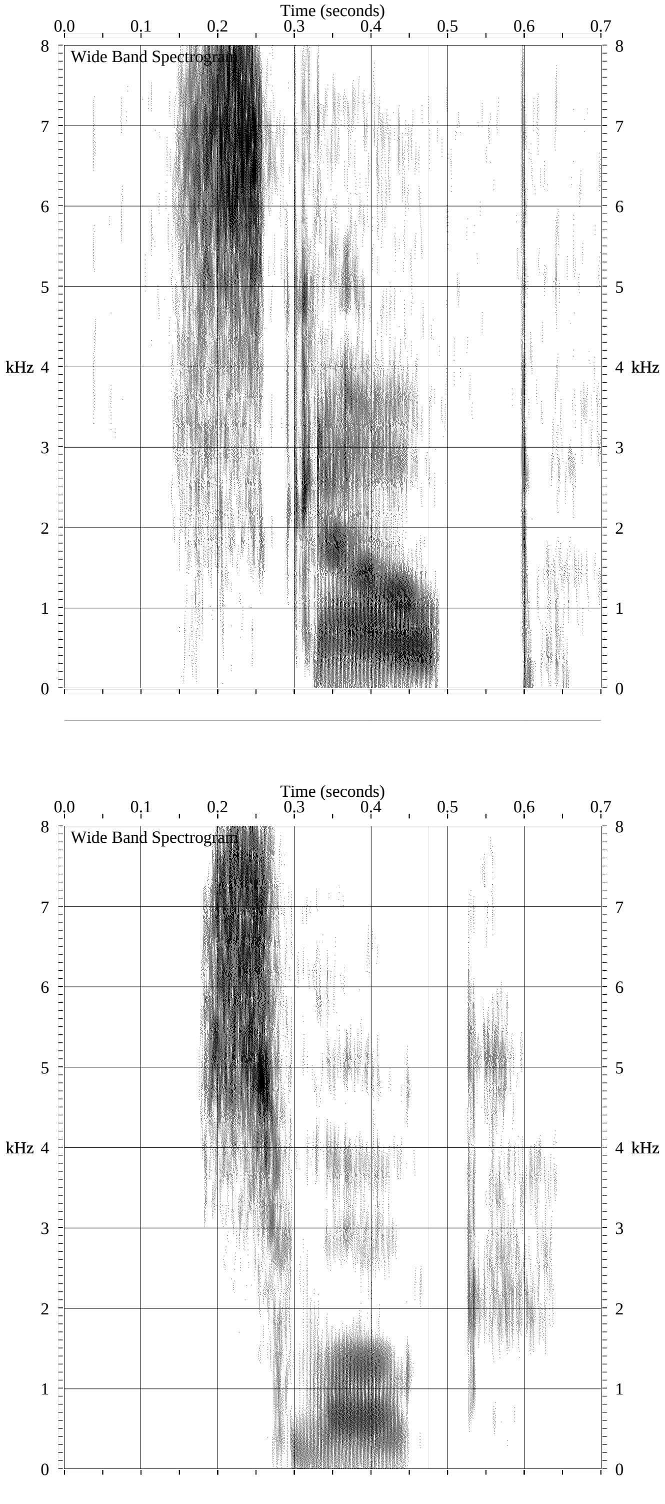


Figure 6-5: Spectrograms of synthetic "scope" and "smoke."
"scope": synthesized from [sk]ylights n[ope].
"smoke": synthesized from [s]wastika nons[mo]king dactyli[c].

6.3.4 Duration lengthening

In this subsection, we examine the minimal pair of “bent” and “bend” to demonstrate how the transition cost function is correctly capturing lengthened sonorants when followed by voiced stops. In Figure 6-6, we see the mono-syllabic word, *bent* /b ɛ n t/, synthesized using sub-word units from the three words: *bellyaches*, *augments*, and *lieutenant*. Let’s examine the mechanics of the underlying search decisions:

The /b/ selected from *bellyaches* is optimal, because both silence and /ɛ/ are correctly matched on the left and right sides, respectively. Because we are leaving the /b/ in *bellyaches* and transitioning into the /ɛn/ from *augments*, we incur a slight transition cost. The cost is only slight, because within a syllable, obstruent-vowel transitions are fairly low cost. The unit cost of the /ɛn/ selected from *augments* is not bad. At least, labial /m/ is on the left side, and the vowel should be nasalized due to the /n/ on the right side. The right side of /ɛn/ is correctly matched with the /t/ phoneme. Next, as we transition from the /ɛn/ in *augments* to the final /t/ in *lieutenant*, we again incur a slight cost, yet the cost is again slight, because within a syllable, nasal-obstruent transitions are fairly low cost. Finally, the /t/ phoneme obtained is optimal, because /n/ and silence are present on the left and right sides, respectively.

In Figure 6-6, we also see the mono-syllabic word, *bend* /b ɛ n d/, synthesized using sub-word units from the three words: *bellyaches*, *draftsmen*, and *brand*. Let’s examine the mechanics of the underlying search decisions:

Up to *draftsmen*, the synthesis decisions are as before. Then, we incur a slight transition cost, because we are leaving the /b/ in *bellyaches* and transitioning into the /ɛn/ from *draftsmen*. The cost is only slight, because within a syllable, obstruent-vowel transitions are fairly low cost. Second, the unit cost of the /ɛn/ selected from *draftsmen* is not bad. At least, labial /m/ is on the left side. Acquiring silence on the right side gives a class (voiced) match. Because the right side of the nasal unit cost function approximates silence and voiced stops to be in the same class, a nasal /n/

with lengthened duration will be correctly selected. But, we should point out that a nasal is lengthened for two different reasons under these two different contexts: In the first case, pre-pausal lengthening occurs, because the source words were recorded in an isolated fashion. In the second case, a nasal preceding a voiced stop is lengthened [12]. Nevertheless, we have managed to select a nasal with lengthened duration!

Next, as we break from the /n/ in *draftsmen* to the /d/ in *brand*, we again incur a slight transition cost, yet the cost is again slight, because within a syllable, nasal-obstruent transitions are fairly low cost. Finally, the /d/ phoneme obtained is optimal, because /n/ and silence are present on the left and right sides, respectively.

In these two synthesis examples, *bent* and *bend*, the nasal /n/ and the vowel /ɛ/ (to a lesser extent) are selected with reasonable durations in both cases. This is owed to the nasal unit cost function which considers amongst voiced and unvoiced obstruents on the right side. Voiced stops, for example, tend to lengthen sonorant regions within the same syllable [30]. Without such a nasal unit cost on the right side, a nasal of lengthened duration could be selected when not appropriate. This could lead human listeners to anticipate a voiced obstruent, and even perceive one when in actuality an unvoiced obstruent follows. The key to perceiving the difference between *bent* and *bend* is due not so much to the voicing difference in the stop (/t/ or /d/), but the lengthened nasal and vowel (a secondary effect.)

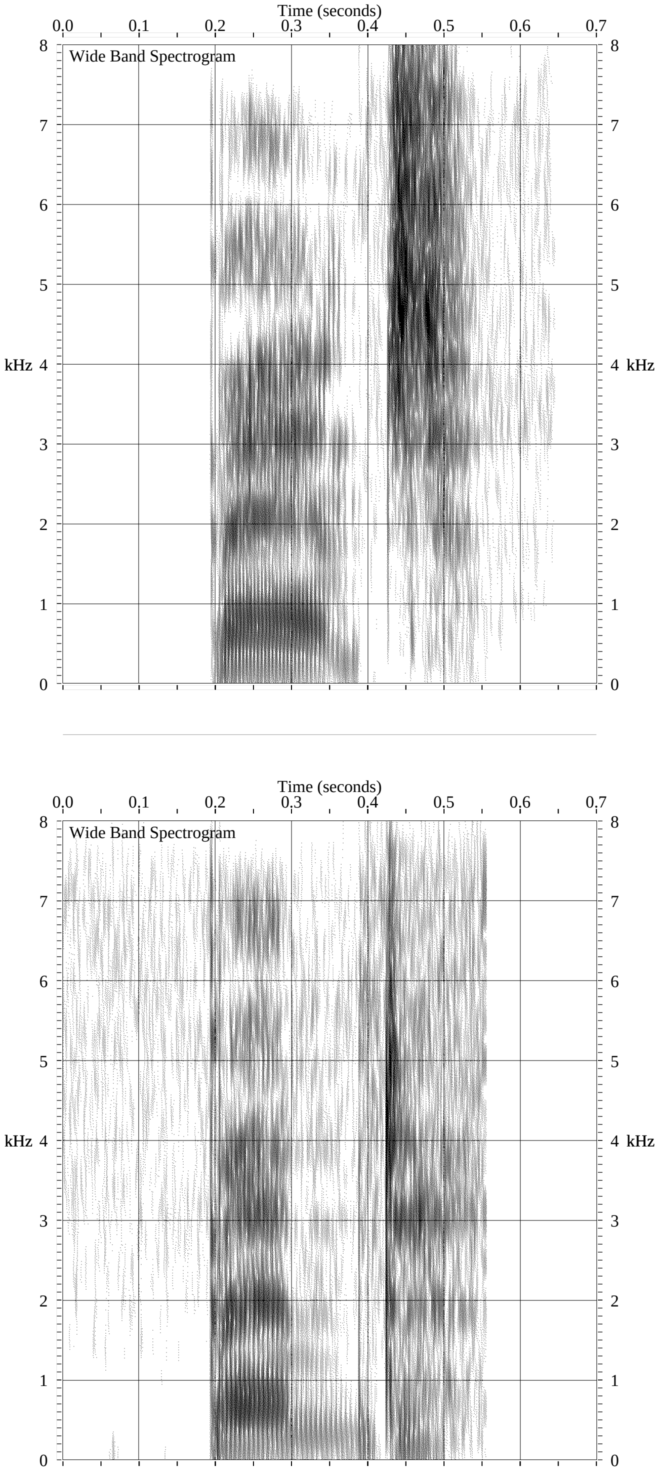


Figure 6-6: Spectrograms of synthetic “bent” and “bend.”
“bent”: synthesized from [b]ellyaches augm[en]ts lieutenan[t].
“bend”: synthesized from [b]ellyaches draftsm[en] bran[d].
Note that the vowel and nasal durations are lengthened in comparison to “bent.”

Chapter 7

Concatenative synthesis experiments

7.1 Introduction

With a concatenative synthesis framework incorporating both sub-word and phrase-level units, we can set out to conduct experiments involving isolated words and full sentence responses. Using sub-word synthesis we will show how proper names can be generated using sub-word units from a corpus of common words. In another example, we will see how sub-word and phrase-level synthesis can come together to synthesize full sentences in which we back off to sub-word synthesis for novel words. Along the way, we will examine various preparation steps and developmental tools that assist the process.

7.2 Sub-word experiments

Sub-word synthesis is useful for the creation of novel words. Given a corpus of words, it can be used to generate more vocabulary from the pre-existing corpus. In this section, we shall use sub-word synthesis to synthesize city names from a corpus of common, non-foreign English words. This demonstrates the decoupling of the two processes of corpus design and unit selection as our training and testing data sets comprise two disjoint domains: proper names and common words.

The task of synthesizing city names is an interesting one and, as city names are proper names, fits our task of synthesizing proper names. Because city names can often have foreign etymologies, this can raise interesting issues when synthesizing from a database of common words. However, it should also be noted that some of the common words of a given language may have originally been imported from elsewhere.

7.2.1 Test data set

We obtained a list of 485 cities from a weather information domain, JUPITER, [34]. All of the pronunciations for these city names were available in the full PRONLEX lexicon. These cities comprised the test data set.

7.2.2 Training data set

To determine of set of words to record which covered the units in these cities, we ran the selection algorithm using the non-foreign subset of PRONLEX. This process resulted in 318 common words, which comprised the training data set, shown in Appendix B. We should point out that normal development philosophy does not permit knowledge of the test data set when designing the training data set. In practice, the training data would consist of the 1,604-word corpus shown in Appendix A. However, that corpus may not necessarily cover all of the city names, as some cities have foreign pronunciations.

7.2.3 Corpus preparation

Both the training data and test data sets were recorded by a native English female speaker at 16 kHz. Although the recording of the test data set is not necessary, it was performed anyways to have an example of each city name actually being spoken by a human. This can be used later as prosody reference templates for prosody modification, as it provides us "perfect" prosody contour.

As the Viterbi search and synthesis process operates on a string of phonemes, it was necessary to phonemically transcribe the training data. This was accomplished by performing forced-path transcriptions using SUMMIT, a segment-based phonetic recognizer [10]. Then, the phonetic labels were collapsed back into phonemic labels (e.g., stop closure and release collapsed into stop) to better match PRONLEX pronunciations. Another set of transcriptions with syllable boundary annotations were also automatically prepared with a rule-based syllabification algorithm. Finally, using the *formant* tool from the *XWAVES* suite of speech software [7], we automatically determined the fundamental frequency (F0) and the probability of voicing at 10ms intervals. To extract prosody contours from the test data, we also phonemically transcribed and syllabified the test data.

7.2.4 Sub-word synthesis examples

In Figure 7-1, we see the city name "Acapulco" synthesized from the words: [a]cclamations, tele[co]nnect, [p]oorhouse, f[ul]crum, pe[koe]. The square brackets denote which sub-word portions were selected. The syllabification of "Acapulco" is: (æ) (kə) (pʊl) (ko^{w}). For comparison, Figure 7-1 also shows "Acapulco" actually spoken by the same female speaker. Note the glottal stop at the onset in the actual word.

In Figure 7-2, we see the city name “San Francisco” synthesized from the words: [s]achet, t[an]credo, bel[f]ry, f[ranci]scan, fri[sk]ier, and pek[oe]. The square brackets denote which sub-word portions were selected. The syllabification of “San Francisco” is: (sæn) (fræn) (sɪs) (ko). Again for comparison, Figure 7-2 also shows *San Francisco* actually spoken by the same female speaker. Note the duration of /fræn/ in the synthetic version is longer than the actual version.

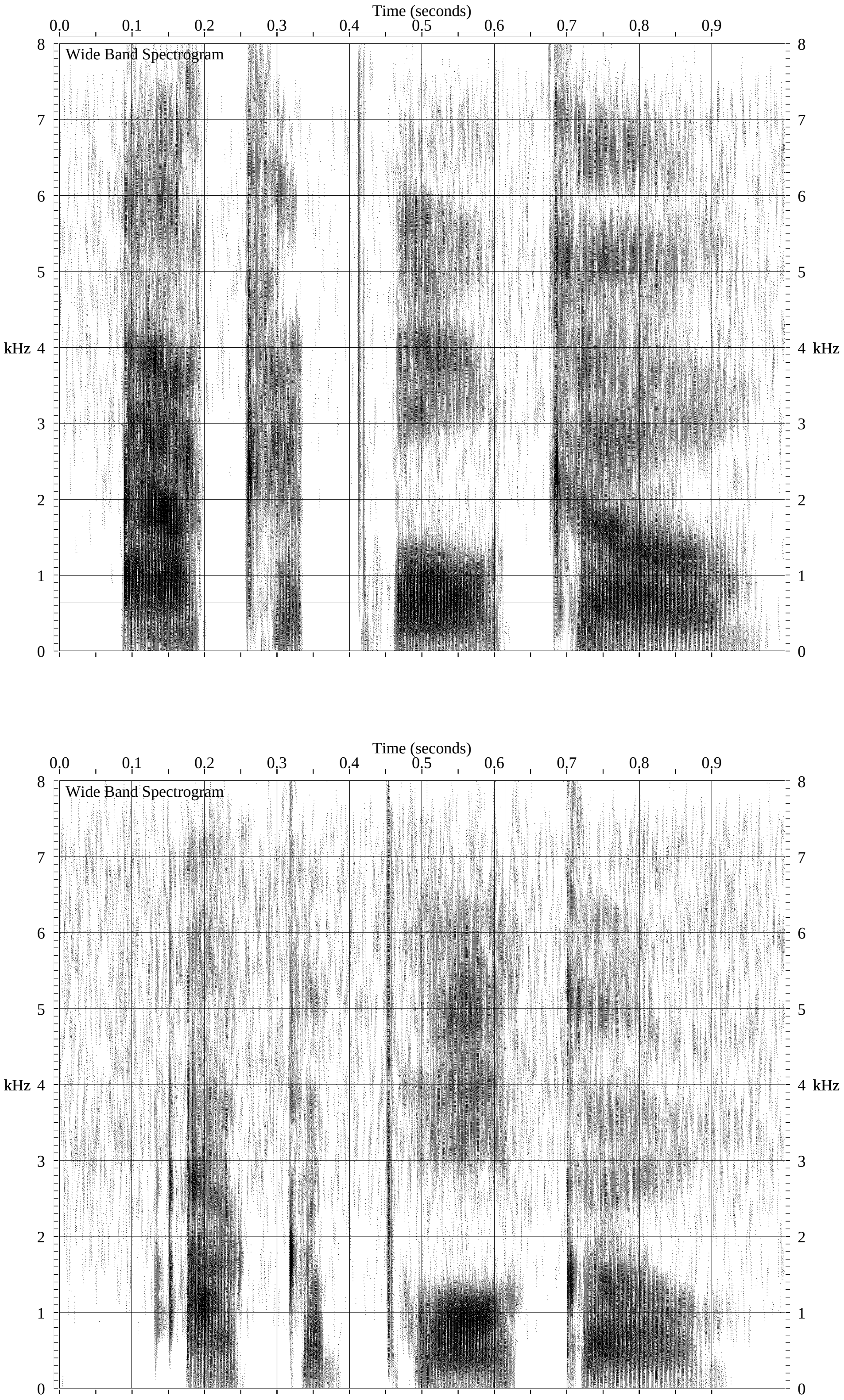


Figure 7-1: Spectrograms of synthetic and actual "Acalpulco."

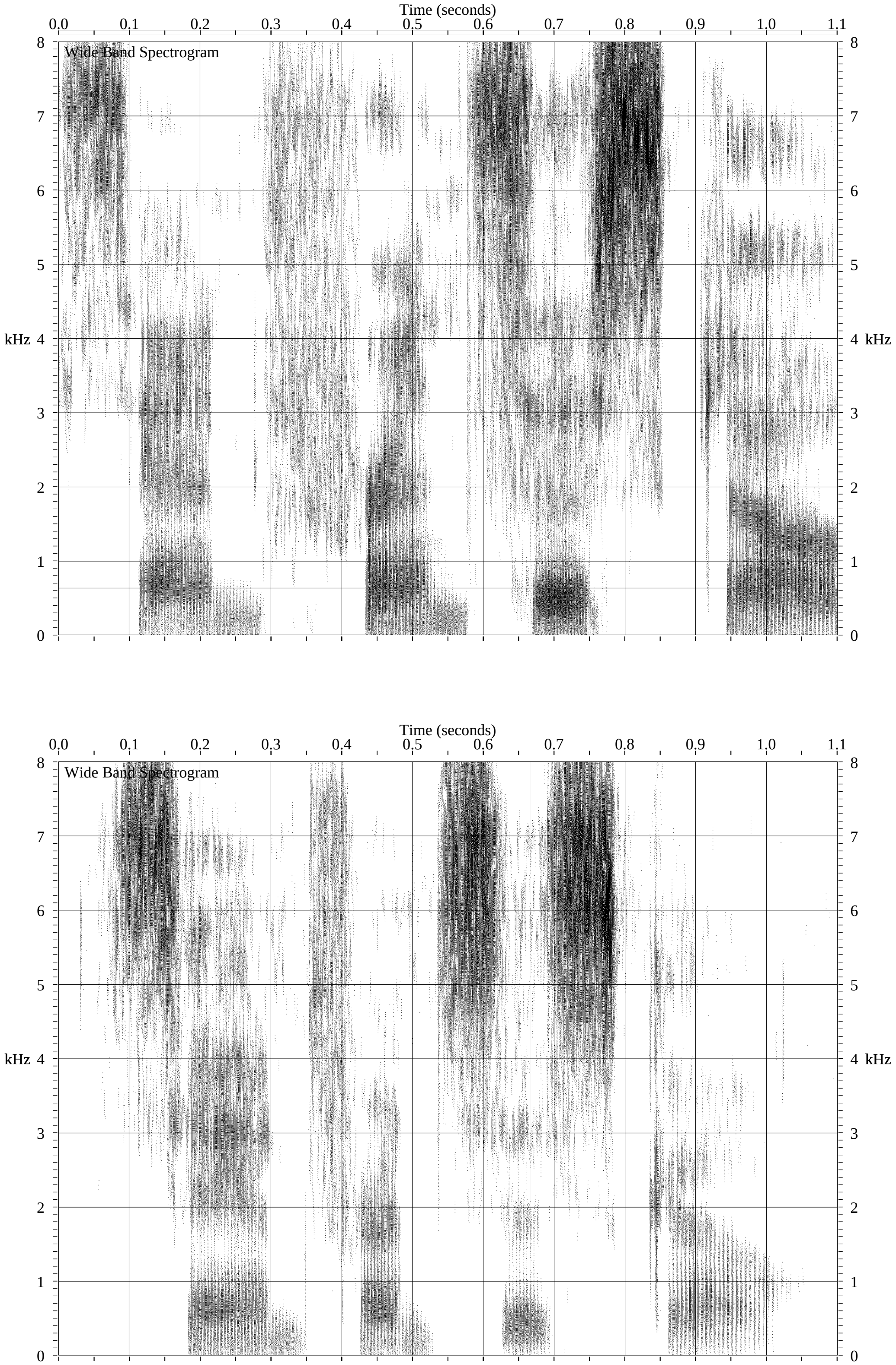


Figure 7-2: Spectrograms of synthetic and actual "San Francisco."

7.3 Development tools

As we performed synthesis of the city names, we would listen to the resulting waveforms, and find the most prominent errors, or unsatisfying regions. Then, we would improve the search metric to fix these cases in a way that would generalize nicely. This process eventually led to synthesis-analysis cycles. We felt the need for tools that would expedite these debugging cycles. Hence, we developed three tools for the creation, viewing and grading of waveforms.

7.3.1 English word synthesis tool

To provide a tool for synthesizing any word from the 68,000-word non-foreign subset of PRONLEX, we developed *word_synth*, a simple graphical interface displayed in Figure 7-3. Entering a word activates the following process: 1) Look up pronunciation from PRONLEX, 2) select unit sequence from sub-word corpus, 3) concatenate speech segments, 4) send waveform to viewing tool. The synthesis-analysis cycle is assisted with an automatic means to reload the unit and transition cost matrices which are user-editable. However, if the user edits the unit and transition context class labels, the sub-word database must be re-compiled, and *word_synth* must be restarted.

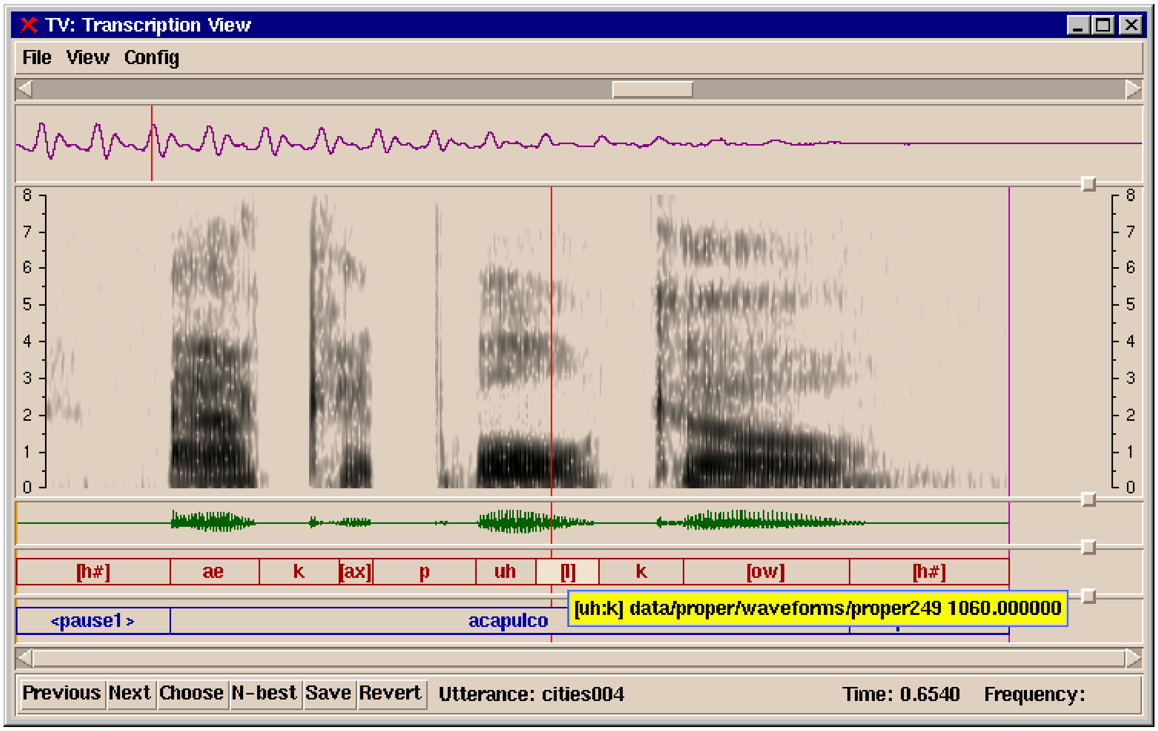
Word: scope

Figure 7-3: Screenshot of English word synthesis tool.

7.3.2 Transcription viewing tool

Starting with the *TV* (transcription view) tool from the SAPPHIRE system [15], we added the capabilities of interactively viewing the search paths from the Viterbi search and the N-best search. In viewing a search path, we could locate from where within the training set certain segments originated. Simultaneously, we could view how the Viterbi path score changed as units were selected.

The original tool displays the waveform, a spectrogram (in this case, a wide-band spectrogram), phonemic transcription, and orthography. In Figure 7-4, we see the modified tool where a popup window conveys contextual information as the pointer highlights different phonemes. This allows the researcher to view the search path in an informative manner. In this example, the /l/ phoneme was chosen from the 249th of 318 common words (*fulcrum*), and matched both the co-articulatory environment and syllable boundary. It is also interesting to note that /ʊ/ came from *fulcrum* as well (zero transition cost between /ʊ/ and /l/), but only achieved a labial class match on the left side. The Viterbi score up to and including /l/ was 1060.0.



[uh:k] data/proper/waveforms/proper249 1060.000000

Utterance: cities004

Time: 0.6540

Figure 7-4: Screenshot of modified tv tool.

7.3.3 Perceptual auditioner tool

As the end-user will be the ultimate consumer of the synthesized speech, it is also important to evaluate how the end-user responds. This is another phase in the synthesis-analysis development cycle. We wanted to create a tool that would allow the simultaneously grading of both subjective and objective dimensions. To this end, we proposed a grading system whereby a subjective measure from 1 (worst) to 5 (best) was assigned to each utterance, and certain objective dimensions could be marked as good or bad. These objective dimensions were: transcription, pronunciation, stress, duration, pitch, and other. This allowed the user the freedom to specify which dimension could be improved.

In Figure 7-5, we see the *audition* tool as it is being used to evaluate utterances. The interface is simple, and the subjective grade can be entered either through keyboard or mouse. Utterances are presented in random order, as the user steps through. In this example, the user is grading *acapulco*, a high score of 4, while mentioning that stress and duration could be improved.

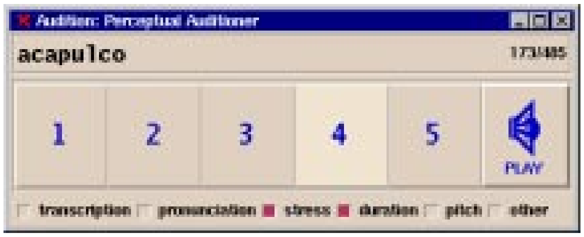

Audition: Perceptual Auditioner
acapulco
173/485
1
2
3
4
5
PLAY
□ transcription □ pronunciation ■ stress ■ duration □ pitch □ other

Figure 7-5: Screenshot of auditioner tool.

After a user grades all the waveforms, the grade report can be viewed using the same tool. The auditioner tool will communicate to the tv tool which waveforms to view. For example, if poor waveforms with poor pitch are to be viewed, simply clicking on 1 and selecting pitch will select just the poor waveforms that had at least the pitch marked. Using the grade reports, we were able to debug the search metric in regard to disjoint properties, such as pitch and duration. The overall quality measure allowed us to target the worst waveforms first.

7.4 Full sentence experiments

Our experiments with full sentences demonstrate the integration of phrase-level and sub-word concatenation. The overall sentential stress will be inherited from the carrier phrase. Proper names will be synthesized using sub-word synthesis.

These final examples come from the PEGASUS system, a displayless flight information retrieval system. Using spoken queries, users can inquire about when flights take off and arrive in real-time. Other in-flight information, such as altitude and speed, can be requested as well.

A phrase-level concatenative synthesizer was designed for the PEGASUS in much the same way as was done for the WHEELS system. For example, flight numbers can be synthesized from smaller constituents (e.g., 4695 can be synthesized from 3[4]7, 7[6]41, 8[9]2, and 56[5].) Multiple carrier phrases were designed for the speaking of estimated arrival and departure times, for example. Figure 7-6 depicts the message template for the estimated arrival time message.

```
:est_arrive   :flight :from_airport
   [ pegasus001 35903 49073 1 1 1 is expected in ]
   :TO_AIRPORT :est_landing
:flight       [ pegasus001 0 3157 1 1 1 <pause1> ] :AIRLINE
   [ pegasus001 13351 17561 1 1 1 flight ] :FLIGHT_NUMBER
:from_airport [ pegasus001 24627 27694 1 1 1 from ] :FROM_AIRPORT
```

Figure 7-6: Message templates from PEGASUS.

To demonstrate the execution of these messages, we provide an input semantic frame in Figure 7-7 that produces the following sentence:

> {Continental} {flight} {46}{9}{5} {from} [G][reen][sb][oro] {is expected in} [Hali][f][ax] {at} {10}:{08}{pm} {local time}.

Again, square brackets are used to denote sub-word boundaries, and curly braces are used to denote phrase boundaries.

```
{c speak_status
   :domain "pegasus"
   :fl_tlist ({c flight_item
    :departing "GSO"
    :arriving "halifax"
    :airline "COA"
    :flight_number {q number :2 "46-" :3 "95"}
    :est_landing {q clock :hour "10$h" :minute "08$m" :xm "pm" } } )
   nfl_tlist 1
}
```

Figure 7-7: PEGASUS meaning representation example #1.

As a second example, we provide another meaning representation in Figure 7-8 that produces the following sentence:

> {United} {flight} {29} {from} [S][an] [F][ranci][sc][o] {is expected in} [O][s][a][k][a] {at} [3]:[28][pm] {local time}.

```
{c speak_status
   :domain "pegasus"
   :fl_tlist ({c flight_item
    :departing "SFO"
    :arriving "osaka"
    :airline "UAL"
    :flight_number {q number :1 "29" }
    :est_landing {q clock :hour "3$h" :minute "28$m" :xm "pm" } } )
   nfl_tlist 1
}
```

Figure 7-8: PEGASUS meaning representation example #2.

Overall, human listeners thought the system sounded natural. Many full sentences were found to be preferable over those generated by *DECTalk*. Most mono-syllabic words were natural-sounding, which indicates to us that the co-articulatory constraints were functioning pretty well. Poly-syllabic words often lacked natural-sounding intonation, which should be addressed in future work. The top 5% of city names were found to be as natural-sounding as fully recorded words from the carrier phrases. The bottom 5% of the city names were noticeably degraded. The co-articulation in both cases were natural, but degradation in the latter could be attributed to unnatural rhythm and pitch. This could be possibly mitigated by having more examples of each unit in the sub-word corpus for prosodic variety, or by applying speech modification methods after the fact.

Chapter 8

Conclusions

8.1 Discussion

This thesis work has four types of contributions: a framework for concatenated synthesis from meaning representations, principles about sub-word unit design for concatenative synthesis, sub-word corpus design, and software tools for synthesis analysis and evaluation.

First, we carried out preliminary work involving phrase-level concatenation from meaning representations. By working with more than just a surface representation, we were able to embed prosodic cues at the semantic level. This demonstrated that at a macro level, maintaining prosody constraints is more important than co-articulatory constraints across the phrase-level constituents. Later on, we saw that at a micro level, maintaining co-articulatory constraints becomes more important within words.

We conducted a set of perceptual experiments to learn where the speech signal can be spliced with minimal perceptual distortions. Places of source change, abrupt articulator movements, and low energy levels are three examples of where speech segments can be substituted. These constraints were integrated into a distance metric for a unit selection algorithm which efficiently searches a database of units for optimal unit sequences using a Viterbi search.

The proposed unit boundaries learned from the perceptual experiments were used to automatically enumerate sub-word units in non-foreign English words. Vowel and semivowel sequences were considered in trigram context with the help of place of articulation classes compressing the consonant contextual information. These multi-phoneme sequences comprised the bulk of the sub-word units. We recorded a sub-word corpus whose prompts were automatically selected to compactly cover these sub-word units.

Finally, the total framework encompassing both phrase-level and sub-word concatenation was implemented in software. The phrase-level component drew on the generation capabilities of GENESIS. Operating in a networked GALAXY configuration, ENVOICE servers return speech waveforms to clients presenting meaning representations as input. This has been deployed within two GALAXY domains: WHEELS and PEGASUS, the latter utilizing sub-word concatenative synthesis. The performance is agreeable, and can be accelerated well past real-time with further work. Tools for word synthesis, synthetic waveform and search path viewing, and perceptual evaluation were created for developmental purposes.

8.2 Future work

In this thesis work, we have made several compromises and simplifications along the way. These decisions lie in the areas of unit design, concatenation splice points, prosody matching, search metric tuning, and the general implementation.

Regarding unit design, we neglected to investigate how consonants should be covered, because of the two orders of magnitude in the consonant and vowel/semivowel set size. It is worthy to investigate the selection of words based on the virtue of consonant sequences they contain. Also, we choose an operating point of a vowel and semivowel sequence length of 2. If we operated with sequences of length 3, we could cover 97.1% of the units and 93.3% of the non-foreign English words as seen from Table 4-8. As alluded to, it may be possible to manufacture VSV constituent VS and SV components. In our enumeration, we also observed that sequences of lengths longer than 3 mostly contain schwas, an unstressed vowel where splicing could possibly be performed while maintaining naturalness. An example of the creation of multi-phoneme sonorant sequences from shorter constituents is presented in Figure 8-1. Here, the city name - "Amarillo" (æ) (mə) (rɪ) (lo^w) - is synthesized from the following words: c[am]ouflage, g[ori]llas, sa[llo]wed. The sonorant sequence of length 5, /ə r ɪ l o^w/ (VSVSV), is being concatenated from /ə r ɪ/ (VSV) and /l o^w/ (SV)!

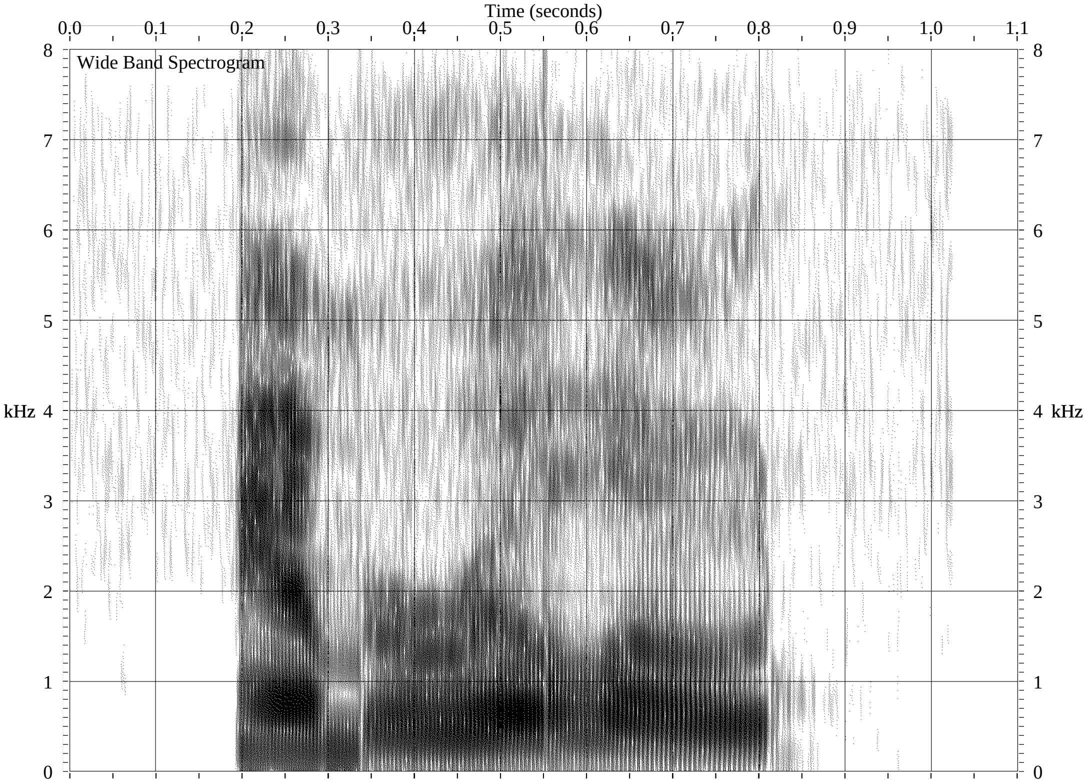


Figure 8-1: Spectrogram of "Amarillo"

There are always more concatenation splice points to examine. For example, besides what has been presented in this work, it may also be possible to break the speech signal in areas of extreme articulation or low energy. For example, one phoneme that exhibits both these attributes is /w/. It may be possible that energy dips outline places where splicing can occur without much perceptual distortion. By driving the unit boundaries out to these articulation extrema, we are insulating the unit interior which contains co-articulation motion. The synthesis units are then conducive to simple concatenation, because of this insulating sheath.

Although we provided hooks for prosody matching in the unit selection algorithm, we did not investigate it experimentally. In the unit cost function, it would be useful to match the prosody of the unit (e.g., is the average pitch, duration, or energy close to what is requested?). Across units, the transition cost function could ensure prosodic continuity (e.g., do the pitch, duration, and energy contours connect across unit boundaries?). In order to perform prosody matching, a prosody contour must be provided. Prosody contours could be obtained by exhaustive recording of a human actually speaking the word, or it might be possible to generate them probabilistically in a hierarchical fashion [6].

As carrier phrases capture prosody, it is natural to speak of the prosody of their constituent parts. Work by others have suggested the existence of *intonational phrases* and delimiting *boundary tones* [26]. These intonational phrases and boundary tones could be used to define *phrase-continuing* and *phrase-final* positions, or tones. Phrase-final tones are often characterized by large final F0 excursions, as well as pre-boundary lengthening of the final syllable, sometimes referred to as pre-pausal lengthening. Such phenomena are generally absent at the phrase-continuing tones.

In response generation, the message templates have certain slots that are filled in at generation time. If information about phrase tones could be encoded into both the dynamic place-holder and the vocabulary, then there would be a good match of prosodic environment upon generation. In the case of the WHEELS system, even when various variables about the automobile at hand change, a carrier phrase would have consistent prosody by virtue of matching phrase tones.

In the development process of the sub-word framework, we found that the prosody of poly-syllabic words could be very unnatural. We partially mitigated this effect by creating labels for reduced vowels as lexical stress markers were removed from PRONLEX pronunciations. However, the natural synthesis of poly-syllabic words may require a sub-word corpus explicitly designed with lexical stress taken into consideration. Examining stress phenomenon at a metrical foot level may also be helpful [3].

Even if the process of prosody matching is entirely bypassed, prosody modification algorithms operating on the speech waveform after the fact may be able restore the intonation back to a “truth” template spoken previously by a human [32]. Prosody modification could also be performed on individual words formed by sub-word concatenation for a better embedding into carrier phrases.

The combining of the unit and transition cost into a search heuristic was ad-hoc. The numbers in the cost matrices were arbitrary, and just manually tuned. The weighting of the co-articulatory and prosodic components within the unit and transition cost functions was also ad-hoc. There are various weighting methods that could be explored. For example, the weights could be automatically learned using gradient descent methods over an error metric. The various quantities that are combined by addition all possess different ranges and statistics (means and standard deviations). Thus, it may be useful to combine the scores in a normalized space of zero-mean and unity standard deviation random variables.

In the general implementation within GALAXY, carrier phrases were designed by hand. Given a range of possible sentences to speak, a context-free grammar could be applied over the sentences for semi-automatic reduction into carrier phrases. Also, the pronunciations for city names were looked up from a static lexicon. Letter-to-sound algorithms exist to automatically perform the grapheme-to-phoneme conversion [25]. As GALAXY is a multi-lingual conversational system, the extension of the ENVOICE framework to other languages also presents itself with interesting research topics.

These five research directions are all substantial projects in themselves. It is hoped future research in these areas will continue to improve naturalness, while achieving higher sentence and vocabulary flexibility in concatenative speech synthesis.

Bibliography

[1] Eleonora Cavalcante Albano and Patricia Aparecida Aquino. Linguistic criteria for building and recording units for concatenative speech synthesis in brazilian portoguese. In *Proc. Eurospeech '97*, pages 725–728, Rhodes, Greece, September 1997.

[2] A. W. Black and N. Campbell. Optimising selection of units from speech databases for concatenative synthesis. In *Proc. Eurospeech '95*, pages 581–584, Madrid, Spain, September 1995.

[3] N. Campbell. *Talking Machines: Theories, Models, and Designs*, chapter III.a, Syllable-based segmental duration, pages 211–224. Elsevier Science, Amsterdam, Holland, 1992.

[4] N. Campbell. CHATR: A high-definition speech re-sequencing system. *Acoustical Society of America and Acoustical Society of Japan, Third Joint Meeting*, December 1996.

[5] F. Charpentier and E. Moulines. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. In *Proc. Eurospeech '89*, volume II, pages 13–19, Paris, France, September 1989.

[6] Grace Chung and Stephanie Seneff. Hierarchical duration modelling for speech recognition using the ANGIE framework. In *Proc. Eurospeech '97*, pages 1475–1478, Rhodes, Greece, September 1997.

[7] ESPS/waves+, URL http://www.entropic.com/products.html.

[8] James L. Flanagan. *Speech Analysis, Synthesis, and Perception.* Springer-Verlag, Berlin, Germany, second edition, 1972.

[9] G.D. Forney. The Viterbi algorithm. *Proceedings of the IEEE*, 61:268–278, March 1973.

[10] J. Glass, J. Chang, and M. McCandless. A probabilistic framework for feature-based speech recognition. In *Proc. ICSLP '96*, volume 4, pages 2277–2280, Philadelphia, PA, October 1996.

[11] J. Glass, J. Polifroni, and S. Seneff. Multilingual language generation across multiple domains. In *Proc. ICSLP '94*, pages 983–986, Yokohama, Japan, September 1994.

[12] J. R. Glass. Nasal consonants and nasalized vowels: An acoustic study and recognition experiment. Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, December 1994.

[13] D. Goddeau, E. Brill, J. Glass, C. Pao, M. Phillips, J. Polifroni, S. Seneff, and V. Zue. GALAXY: A human-language interface to on-line travel information. In *Proc. ICSLP '94*, pages 707–710, Yokohama, Japan, September 1994.

[14] C. Hamon, E. Moulines, and F. Charpentier. A diphone synthesis system based on time-domain prosodic modifications of speech. In *Proc. ICASSP '89*, pages 238–241, Glasgow, Scotland, May 1989.

[15] L. Hetherington and M. McCandless. SAPPHIRE: An extensible speech analysis and recognition tool based on tcl/tk. In *Proc. ICSLP '96*, pages 1942–1945, Philadelphia, PA, October 1996.

[16] X. Huang, A. Acero, J. Adcock, H. Hon, J. Goldsmith, J. Liu, and M. Plumpe. Whistler: A trainable text-to-speech system. In *Proc. ICSLP '96*, pages 2387–2390, Philadelphia, PA, October 1996.

[17] A. J. Hunt and A. W. Black. Unit selection in a concatenative speech synthesis system using a large speech database. In *Proc. ICASSP '96*, pages 373–376, Atlanta, GA, May 1996.

[18] R. Kassel. Automating the design of compact linguistic corporation. In *Proc. ICSLP '94*, pages 1827–1830, Yokohama, Japan, September 1994.

[19] R. Kassel. *A Comparison of Approaches to On-line Handwritten Character Recognition.* PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, June 1995.

[20] C. McLemore. COMLEX English pronunciation dictionary. URL http://www.ldc.upenn.edu/ldc/catalog/html/lexical_html/comlexep.html.

[21] H. Meng, S. Busayapongchai, J. Glass, D. Goddeau, L. Hetherington, E. Hurley, C. Pao, J. Polifroni, S. Seneff, and V. Zue. WHEELS: A conversational system in the automobile classifieds domain. In *Proc. ICSLP '96*, pages 542–545, Philadelphia, PA, October 1996.

[22] Douglas O'Shaughnessy. *Speech Communications: Human and Machine.* Addison-Wesley Publishing Company, New York, NY, 1987.

[23] L. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals.* Prentice-Hall, Englewood Cliffs, NJ, 1978.

[24] E.O. Selkirk. The syllable. In *The Structure of Phonological Representations (Part II)*, pages 337–385. Foris, Dordrecht, Holland, 1982.

[25] S. Seneff, R. Lau, and H. Meng. ANGIE: A new framework for speech analysis based on morpho-phonological modelling. In *Proc. ICSLP '96*, pages 110–113, Philadelphia, PA, October 1996. URL http://www.raylau.com/icslp96_angie.pdf.

[26] S. Shattuck-Hufnagel and A. Turk. A prosody tutorial for investigators of auditory sentence processing. *Journal of Psycholinguistic Research*, 25, 1996.

[27] F. K. Soong and E.-F. Huang. A tree-trellis based fast search for finding the n best sentence hypotheses in continuous speech recognition. In *Proc. ICASSP '91*, pages 705–708, Toronto, Canada, May 1991.

[28] K. Takeda, K. Abe, and Y. Sagisaka. *Talking Machines: Theories, Models, and Designs*, chapter I, On the basic scheme and algorithms in non-uniform unit speech synthesis, pages 93–105. Elsevier Science, Amsterdam, Holland, 1992.

[29] J. Terken and R. Collier. *Speech Coding and Synthesis*, chapter 18, The Generation of Prosodic Structure and Intonation in Speech Synthesis, pages 635–662. Elsevier Science, Amsterdam, Holland, 1995.

[30] J. P. H. van Santen. *Speech Coding and Synthesis*, chapter 19, Computation of Timing in Text-to-Speech Synthesis, pages 663–684. Elsevier Science, Amsterdam, Holland, 1995.

[31] J. P. H. van Santen. Combinatorial issues in text-to-speech synthesis. In *Proc. Eurospeech '97*, pages 2511–2514, Rhodes, Greece, September 1997.

[32] J. Yi. Time-Domain PSOLA concatenative speech synthesis using diphones. Bachelor's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, May 1997.

[33] V. W. Zue. *Acoustic Characteristics of Stop Consonants: A Controlled Study.* Sc.D. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 1976.

[34] Victor Zue, Stephanie Seneff, James R. Glass, Lee Hetherington, Edward Hurley, Helen Meng, Christine Pao, Joseph Polifroni, Rafael Schloming, and Philipp Schmid. From interface to content: Translingual access and delivery of on-line information. In *Proc. Eurospeech '97*, pages 2227–2230, Rhodes, Greece, September 1997.

Appendix A

Recording prompts for non-foreign PRONLEX coverage

congressionally nonrenewable langur whomever niagara unremarkable niobium pneumonia slanginess trachea fulmar buprenorphine cardiopulmonary limelight wormwood rainier overenthusiastic gargantuan loanloss overanxiety channelview myopia fulcrum livelihoods oklahoman sauerkraut flamenco cacao foulmouthed nonlinear do-it-yourselfer longrange ringworm elbowing weeklong linear's aquaculture compuserve linguae families costumier ergonomics erythropoietin autoimmune tourguides goldblum's scorpio bullhorns tapioca ennui diametrically monosyllabic tokio armrests clothier powerpoint multiemployer gunlicks geometry melba ammoniated foolproof unrehearsed eelpout transohio rodeos babysittingwise winceyette kielbasa ecumenical seniornet genre vivaldi's spermatozoa dossier cameo smallpox islamization nearby equimark kafkaesque spokewoman minutiae unreconstructed psychiatric stymieing amniocentesis counteroffensive rawboned pipework commingling compugraphic unwinnable innermost juveniles decoying spontaneity cover_ups milner centrifugal sunray shampooer somewhat diurnal smallcap wembley thermionic neonatal empowerment glucose cellmates avowedly yemenite geiger raucous unromantic semiannual affixing conjunctivities deemphasizing repealment banal surrogates angiomedics clawback symbiosis unratified unnerving competitor enjoyer elastomers unlocated sublime chowperd yaw yo uhoh law row nah yeh ye wah wa rah woo oil low rue wow hmmm who'll maw now de moo mayor bra sly churl mull mayo ulterior mar duo coir soya bayou i'ers oomph flaw craw d'ye bio ne'er hurl gayer annoy slew coyer shall payee foyer schwa haulm bowie neo ilk gooey yep yaws newer aon meow yeas yip moll won't moor quo yipe mawr weren't meows biannual baaing payees snowy churls bayous gnawing cayenne buoys inlay overawe guying swam cowing swung sayeth slain cooing swum hoofs foyers lingerie slurp wound paranoia coequal knolls squaw choate neighing molls kelp joyous mooing cawing mowing showoff boings cairn fayette judeo tibiae coyest ireful malmsey yiddish skiers ennuis faience unwed koumiss inwrought gurkha annoying denier aplomb however raunchy kruger overate unwound messiah fullness cairns sequoias coolness highest sprung dearness denying unarmed mayonnaise dalfen tracheae lingua showiness sapwood tunneling dullness ginseng cacaos stingray myopic longwood olfactory cognac direful foulness nowadays cameos raiment paleness unweaned gloaming guileful meiosis poorness unlatched socalled nearness sourness longlegs snowcapped joyousness myocardial dismaying smallness staleness fulmars stillness mormons stingrays meridional oneself unlaunched so-and-so singable diaphanous enjoyable subpoenaing cardiologist unruffled conduits insinuate judaism unwonted picker-uppers nonutility unknowable overanxious systemwide defilement semiotics intelligentsia multicenter superoxide stoicism costumiers nonalcoholic intelligentsias overambitious unlamented spermatozoon unairconditioned thiodiglycol cromlech obloquy neanderthals gomphrena sawmill schoolgirl underemployment yahoo scarecrow whatsoever trimetrexate blameworthy nainsook strongroom hoopla runways wigwam entrepreneur renewing disunion ploughboy trochaic fireclay gremlin tramway lawnmower primrose one-on-one piano wavelength nonunion mugwump liqueur lapwing goalkeeper fairground springlike fearfulness workroom circumlocution albani aquamarine dreamlike baobab mainlanders toolmakers whammy ipse hangout mailgram rationale woogies lemur gaga foundering sovietologist hourglass radii shopwindows papaya cucumber swamis ahold demoniacal beguile judeochristian sonar malnutrition tapeworm accusation rawhide almoners aftereffects unrepresented overachiever moustache semiautomatic deprenyl atlas assemblymen eyeopening jerboa rye ray wee lie our ley roe owl whoa yea lee coy the wowed miaow cowl naw phooey woad pol kiel hoey null yon moil woke dull yen wrought aisles wok yak awing mile mare lung pal womb eon roisterer mayor's naive keeled sahib payout buyout palms mulberry mowers europewide blurb buoying doyen iota quern pawing quoin poem payoff hillbilly coors's bloke mares toying pals sloop biennial chaos boas hookah illness coyotes overawes oilman bonker wafts hyena seance oasis trousseau duenna penile simoom kiosk unearned inroad skillfully crux sequoyah dowager moiety peyote gaiety cowman dinghies bongo hairnet farrago whoever unlawfully paranoiac umlaut ploughmen filenet mailmen doubloon doable beerman hiatus virtuoso coeval poetic beguiled mucus siesta ailment dolmen underage hemlock sciatic someone tollgate biopsy howitzers quango phaeton gulden sunroof annoyance sunlamp toolbox unleveraged songwriting stopwatch unaltered squeamish ointment ringleader bitumen gearbox diameter coxswain unlabelled cameramen chasuble diocesan annulment unlimited signalman coefficients psychoanalysts fiberoptic installments exhumation disfigurements entrepreneurs heehaw ulnae wholemeal molehill brouhaha gourmet wrongdoer emu quagmires pimpernels onlooker unenviable rapiers macrobiotic airmail scaremonger cuneiform hungover rehired rococo goodlooking multihull therefore enough's ramrod unannounced brainwave croupiers grenadier loquat oilcloth wampum behavior oilfired snowplow quinine scrawniest penguin sauvignon macaw wonkier kowtowing unluckiest ethanol reschedule unwittingly diagonals bang_up grandeur ignore upward heatwave youngish widowers appliqueing recoil semifinalist clawhammers demimondaine coeducation companionway outlaw millboard courgette err oi whir y. whey i'll lieu r my gower dowel slow showy gear sewer dewy knoll yum owls rung owing mell rune hulk meals naiad dais talc moors moiled wildlife fiat cohen booing wont prong slung wifelike kayak mauver geeks yank neon enroute cowgirls inlays raceme octroi online nuance abrupt cyanuric quagga wounds pelmet zoophyte overeager unwind firemen sunroom shamrock riffraff between toymaker dearment tramways gunrunner counteroffer psychiatry mimeograph endowment inculcates humanitarian pueblo cloisonne nailfile yeomanry leukemia slovenliness flyblown sharecropper loamier summerfare gonging pioneers parfums stringier bohemian monkeying charcoal monthlong memsahib miasma biopharmaceutical bluegreen diaphragm coypu naivete halberdiers oceanic livelong mutinying oversimplifying rowntree's carbonell tangiest lowbrow refugee overabundant split_level courtiers yucky overkill boycott aglow repayable yeastier salsify supremacist cuckold browbeaten polecat powerboat impairment a ow l ire eel hours whoop loin pearl ploy glue lurk eyeing gal uhuh woos yap dieu lain pry kneel gung maim cry dayer towelled prone swain mires miaows swoon gleam kneeled toured cluck bowing cowles vehicular clerk swap unripe slake quoit plonk swoop pleb lorgnette unlock unloose stoic unlearn flaunt swept anaemia earmuffs maunder enclave squabbling biomass tailgates comrade vulnerable andiron bivouac canals bayonet homeliness fickleness clement dioxin unrequited deniable girlfriend superimpose neocolonialism disunite biographic overestimate walkway chapelgoer papaw lakewood loveliness gloomiest twangy ionosphere ropeways brother_in_laws cuckoo elkhound hoho semipro woodworm unreconciled semiquaver prejunior linedraw huckleberry cloakroom promo leviathan waterhyacinth whipsawing bobsleigh cyanamid moslem poltroon nodules blowfly rootless chromium alcohol tourneys parterre turnoff subsoil rissole carefree flamboyance paleface englands yawn woefully yearn iamb earls roof im joyfully frugally tearfully duet coils smog drawn flog slouch gears honk brook annuls brine swan venue trauma googlies onrush deify overeat silkworm foible unleash crofter unloved inlet nonlife inland halfwit permeate bimetallic preretirement diploma fructified coexist killjoy troglodyte clockwork halcyon fingernail air-dry promulgate bouquet hermaphrodite rainwater slipway skyhigh sliminess ironwork lenience cowbell calmly lumbago numskull coalfield archive wingless framework coalmine premier counterespionage biography brownwood southland brickkiln equinoctial footslog tollbar amiable rehearsals watchtower ale thou pow yam keg wop eels mime boor lurch loiterer woof bleak bleep learn yup soil's groin croon bairn tears sayers slope broom crook lounge flamethrower tweaked clown prune enrage maudlin blonde lemon lawmen punctilio anarchy anoint ionized squawk unwise momentary slenderize chairmen uplands mailbag superannuate maltreat contumely strength homeland stalemate duodenum culminate commutator overemphasized underexpose encirclement steelmills bulbul forknowing gurgling aircrew thingumabob dyslexia freudian almanac corncrake frumpier latrine biochem queensway earplugs sleepwalker black_and_blue lengthwise ballgame hyperactive counterattraction yashmak congresswoman oilskin nonrecyclable female's shuttlecock mononucleosis purveyance unemployed u ear ore ormolu nook furled mall gray boyish rhymed swerve slick slim phlegm quake aren't geld slam clung beyond grind pieta crink oncology meander affianced snorkeling mediocre breakwater pilgrims trefoil ophthalmoscope sulkiness aquaplane prefigure schoolboys kleptomaniac tragedienne doornails moorfowl anymore begonias rostra playschool maypole mammals pustule womanlike taxpayer leapfrog clangers mortgagepower baa whang rogue yodeller sleek llama bourse claque coward doers gales corm inaugural coagulate bronchial scarce flounce mango kodiak hemline schoolmen snowmobile barbecue outre crocodile mummify moviegoers prosaic questionnaire swampiest glaucoma trainload fulfilment spikenard full_scale hazelnut weaponless reconfigured whenever gallbladder rupiahs rough_and_ready uninvolved gimmickry trunkline preciosity hernias gainsaying crosswalk moonlight healthcare kenworthy fungoid minicar unreflective ukulele coin wain rookery yes tieing elm loom shoo-in marmoreal dowered rind poet payers moist brawn cranked karma fairview skulk crept hoity_toity bloodlust anarchic wobbler rename bioscience horsehair decor mother-in-law drawdowns leprechaun lobbying chronicler archdeaconries gangways doctrinaires rapacity loonybin tomfool leguminous monoclonal irk cow fray look trough shrine glide murmuring mild drown hind adroit enright delineate psychoanalyze manumit euphonium franklins arpeggio morningstar brand_new lunchroom swashbucklers trembling proclamation remediable dryclean ploughshare malpractice animalcule scofflaws alfalfa remedying enamel dunghills troublous wean lame rake bias slab peon tooled twig ranee awkward creaks unused coinsurance slogan hailstorm peafowls milkweed lanyard dogleg gladewater rutile torpedoing prevail swinger only threadlike shire wake knout coiffeur ain't noon cared cleave grime slug frond inchoate agglomerate geography disgruntlement lawgiver schoolchildren intracranial eardrum bluebells remelded quicksilver polyhemoglobin overindulge sprightliness bradawl lawplan foreplay inadequacy lackluster disarmament billhook hoarfrost kevlar yolk rom cried plume slag clobbering newish scalp squeak marbling quaint falcon forgoing lugsail astronaut redwood epiglottis loophole theocratic caveat swindlers riflemen lousiest gangplank twenty-twenty saleswomen langsyne maestro lockjaw microscopy laws tarn should mercurial flung swob reissues torque swipe mamma granola meres ephemeral snaky skirmish interact calcium procommunist lightbulb flipflopped desegregated humankind battlefront billfold thole we've slowed slewed slaughterer swaggerer clique co-op powered paying glum swift glimpse tailpipe viable drive-thru premature corkscrews recusant lymphoma evensong rug profiled mob grown slob lens drapery pooled pelf brave sweep sulpha quench overexert racoon megadeals ultramarine foxglove twine robbery cleanlier gland elegiac slashed argument wedlock claptrap omelets flagrant roundrobin airfares fireball ranchland lush slosh barn aerodrome puke slum enwrap palmed videotron sucrose fruitcake aigrets northward noisily tires fumes foiled slavey cloudy dyazide manufacturer grumblers occupier poohpoohed couscous route wan primarily shriek drone drachm flim ballpark tortuous ostler troubadour waxwork butyl whim lave rockery likelier roam fluff opera snuggery sled quandary strep guttersnipe playbills umhum napalm rhizome herb cupola shroud shrimp funerary lumberroom salve trunk quadrangular turboprop lampoon rang fay calculus croak freckling wisecrack rum nerve chewing rushlight unanimous annunciates petrostrategies sleaze calk normalize grieved bribery lot fame wine hike paled mauls plumb legless browse quote snoop demarche carnivorous waistline greybeard lodestars chaired clime bulk flab blunt blind fingerprint provocateurs galt gregorian legislate wheat blues fledgling orchestrate wordbook clam crush cream move sweat flocked blench apprehend legally numeric mooch roomer tarp prep alleviate shell claim gulp solve freightways hush_hush greataunts trots grub faun sweetbriar unexploded wren freak cliff's scriptwriter february coupe mound joinery molds off bum floss lose actuate snapshot all's flinch mortuary gripe drench quip theatre souls multiprocessor pulse leisurely gluten laid novelly drudged wherewithals tell's pawed trues swished lass chew user plight raft fleet littler greed puffery quitter tugs

Appendix B

Recording prompts for coverage of 485 JUPITER city names

rehabilitatable overpopulation receivables detoxification abnormalities cytomegalovirus intermarried incubator indoctrination teleconnect unilaterally oklahoman deerstalkers motorway oceanography apprenticeships hypochondria gendarmerie idiosyncrasy numberless behaviorists caribou unpublicized guacamole nickelodeon nonproductive nutritional homophobia get-togethers do-it-yourselfer syllabaries dactylic anarchism lumberjacks jurassic miscalculations laguna hallucinogen recourse bandeaux dunderheads whitetail nebulous multimarket corporatist safeguards coupon lakeshore mobile maestro fairview lockjaw turnoff autoparts rockingham icefloe olympiad heartburn softcore clearwater trashmore warwick lingua philatelic barbecues camouflage lafarge's forecastles draftsmen cesspool lodestone kinky sachet furbelow and-a-half dreamily rococo limo poignantly gratia leftover bargee cardio acclamations quebeckers panorama crucifixes tancredo anglicize darlings antique littleboy firedamp waco's eskimo ablative handwritten comeback peepshow virtu brownout jonesborough kola robots headboard minaret augments pantyhose pekoe bomberg swastika borzoi treadmill franciscan bugles alexandrine gar hues mail polygraph aquacise rosebud elbows oak's propound progress rust overindulge brotherhood allspice shampooed memsahibs clays legwork driftwood flatten code sloth sloppiest enfolds clans rondeaux swain heed twistier richly hotfoot doorbell almost health's fluorine theatricals bellyaches opera pool flume hole burlapped lobbyists you'd flintier truth northfield lionize crisped harrying ramped acquits thoroughgoing sootier borough pawnshop west woodwinds stylized cleaves pueblos hahas wreath fowler spookiest harped yak skylights freckly provosts willpower eat fulfil poorhouse els swan's mall libra gnarled nuclei luxurious rake shoehorns salvo scalp colleague violin curlew laureate wheelon jeweler sole bailee hilt high therein pell haycocks niagara galleria sallowed wherewithal hoagies powell aryan alleyway calculi cayenne moritz marquee island heehaw liqueurs coypus adios transohio judeo erotic welling sirups skyboxes irrigate bureaucracy psychobabble endowment's altruistic skyrockets illogic theocracy ta-da reatta calulai fulcrum theodolites halibut gorillas milliliters hagiology lieutenant demerara southland's declaratory onomatopoeia grandson auditoria imbued proms backcloth slat cusp soiree springier blackwoods plum's accelerandos logs transfix tailored remakes landfills husk comb yeller washbowls lush harlots eft brand serviettes nonsmoking slaked colleen lot's reroute thallium adroitly unworthy rollicking willingness shoplifts belfry remark cogs nope escapes exalts alums cholera zoomed screenplays snagged friskier windpipes outdo corvee dam's needlecraft bistros toothbrush potbound candlewick faults