TR-IT-0054

Chatr Overview *and* Synthesizing English Intonation

Paul Taylor

1994.4.1

ABSTRACT

The first part of the report discusses the Chatr speech synthesis system and the design methodology behind it.

The remainder of the report gives details of the intonational module in Chatr which is based on the new Event/Tilt model of English Intonation. It is shown how Rise/Fall/Connection (RFC) descriptions of Intonation can be re-phrased and described by a theory of intonational events. Next, the dimensionality of the event description is changed from the low-level RFC space to the higher level, more meaningful "tilt" level. The descriptions are further transformed by describing a set of features and labelling the events in terms of those features.

These concepts are discussed thoroughly and performance figures are given at every stage showing the accuracy of each mapping. The report concludes with a description of the synthesis algorithm itself and documentation of the relevant Chatr functions.

Chatr Overview
and
Synthesizing English Intonation

Paul Taylor

April 4, 1994

Abstract

The first part of the report discusses the Chatr speech synthesis system and the design methodology behind it.

The remainder of the report gives details of the intonational module in Chatr which is based on the new Event/Tilt model of English Intonation. It is shown how Rise/Fall/Connection (RFC) descriptions of Intonation can be re-phrased and described by a theory of intonational events. Next, the dimensionality of the event description is changed from the low-level RFC space to the higher level, more meaningful "tilt" level. The descriptions are further transformed by describing a set of features and labelling the events in terms of those features.

These concepts are discussed thoroughly and performance figures are given at every stage showing the accuracy of each mapping. The report concludes with a description of the synthesis algorithm itself and documentation of the relevant Chatr functions.

Overview of Work at ATR

There were two main strands to my year's work at ATR. The first concerned the design and implementation of a real-time speech synthesis system, and the second concerned intonational analysis and synthesis. This section gives a brief overview of the philoshopy of the Chatr system, and the main part of the report gives details of the synthesis module.

Chatr Philosophy

In my view, one of the biggest problems in our field of research concerns programming style and maintainability. Many researchers have no formal computer training, and although competent at designing algorithms, often fail to pay attention to many of the other issues in computing, such as system design, real-time performance, maintainability, ease of use, robustness of code, adaptability and documentation. Although it may not be obvious to the people involved, ignoring these issues has a direct consequence on their ability to perform research and subsequently publish papers. Many researchers would admit that they seem to spend a lot of time messing around with their computers in one way or another, but one often hears the opinion that this messing around is just something you have to do when you use computers. Also, one often gets the idea that there is a large amount of repetition of work, in that someone else has previously written a similar algorithm but it is unusable; maybe because it is lost, because it only runs on one type of machine or because the code was not written in a sufficiently general way so as to be modified. All these things eventually make the writing of software slower which in turn hampers research progress.

Chatr is a speech synthesis system that was designed to overcome as many of these problems as possible. Chatr is a portable piece of code, meaning that it can run on many different machines. Chatr is real-time, so that one can try and test new ideas quickly. Chatr handles all hardware interaction, so that one need only call a subroutine, "play", to hear a waveform; one does not have to have any knowledge of the hard ward details. All file i/o is handled by chatr, allowing waveforms of differing file types to be read in without the user having to worry about particular file types. There is no data held in the code itself, as all data is kept in files. This means that one can change the parameters of a model without re-compilation.

Perhaps the most difficult part of Chatr to design was the central data-structure. Here we adopt a black-broad like architecture that lets module writers store data in an easy to read and easy to write manner. The architecture consists of a number of streams which are defined at run-time. Each stream is a list of cells, with each cell representing a particular data structure corresponding to some linguistic object. For instance, we have a word, a syllable and a phoneme stream in the English version of Chatr. Each cell has a number of relations, which specify the cells in other streams which are related to that cell. For example, syllables are related to words as every word has one or more syllables. This data-structure allows easy algorithm writing as on only need worry about the algorithm itself; all the data is handled automatically by Chatr. The design of this part of Chatr was difficult as the architecture had to store complex data while still being easy to read from and write to. Furthermore, the architecture had to be fast. As the processing of a typical utterance may involve thousands of read and write operations on the architecture, it was important that these operation should not slow down the execution of the program. As it stands, the current version of the architecture is very fast, but improvements could still be made concerning the ease of use. At the most basic level, the architecture is very flexible meaning that one can also represent a data structure somehow and that the architecture is never too restrictive.

The other main feature of Chatr is the Lisp language interface. Some types of data, e.g. waveforms, naturally lend themselves to certain file formats (i.e./ binary). However, for the most part, data can be stored in ascii files. Nearly all data stored in Chatr is in the form of lisp-like bracketed structures. These structures are very powerful as one can usually describe complex data representations such as trees very

easily. Furthermore, all lisp file i/o is handled by a single subroutine insuring reliable reading and writing of these structures.

Chatr also has a command line interactive interface, allowing a user to type in new utterances, and change the system parameters of Chatr without having to stop the program or re-compile any code. There is an on-line help system which allows users quick access to documentation on Chatr functions.

These features make Chatr a pleasant and easy environment to work in, and allow fast and easy writing of code. Because Chatr provides so much low level functionality, one can write algorithms that are truly independent of operating system, machine, and file type specifics. This facilitates portable code. As chatr can be re-configured from within, it is possible to have a wide variety of modules which perform the same basic task. The user can chose which module to use at run time.

Chatr's Status

After one year of development, Chatr has fulfilled all its design criteria and is now a fully functional real-time speech synthesis system. Over the last few months, the quality of the output speech has increased dramatically, and much of this success is down to the ease of use with which new algorithms can be developed.

Most of my research has concentrated on the intonational component, which is the topic of the remainder of this report.

Synthesizing English Intonation

Contents

1 Introduction: The Problem

In the waveform generation part of speech synthesizers it is necessary to specify the F0 (fundamental frequency) values with which the speech should be produced. The intonation component in speech synthesis systems is concerned with what F0 values should be fed into this waveform generation module.

It is clear that providing a sophisticated intonation module for a TTS (text-to-speech) system is not as important as knowing which phonemes to say and how to say them. For instance, a utterance produced with a constant F0 value (the simplest possible intonation module) is still usually intelligible. So why bother with an intonation module at all? First of all, although the relative importance of the intonation component may be less that that of the phonemic component, it is still important. If a synthesizer had a very good segmental component and a very poor intonation component, it would be apparent that it is the intonation component that is making the overall speech sound unnatural. In fact, in my previous work at Edinburgh University, the segmental quality of the diphone synthesizer was such that it was deemed that the poor intonation component was the major source of unnaturalness in the system. Secondly, although again not the principal factor in intelligibility, it has been shown that good natural intonation makes the speech more intelligible overall (Silverman, 1993). If words that should be accented actually receive accents, and questions that typically end in rises actually do end in rises, the speech should be easier to understand. Thirdly, speech synthesis, like the speech field in general, is moving away from the carefully controlled professional speech of the laboratory towards more spontaneous natural speech. The intonational differences between these two extremes of style are arguably greater than for any other aspect of the language, and so it is evident that previous intonation models which cater for laboratory speech will be seen to be inadequate for more spontaneous natural speech. For example, many intonation systems deal only with "neutral declarative" speech, which it should be obvious is not the way people talk in real conversations. Any speech synthesis system that is to be used in a dialogue situation needs a more sophisticated intonation module.

1.1 Why it isn't Like Phonemic Synthesis

In very broad terms, one could split the main part of a speech synthesizer into two parts. The first part takes a stream of words and produces a stream of phones (either phonemes or allophones), and the second part takes these phones and produces waveform, be it by formant, diphone, unit or articulatory synthesis methods. Splitting the task in two is a powerful device as it allows strong modularity of the problem. The first component works in a dialect dependent way, and is oblivious to the speed and style with which the utterance is to be produced. The second component need only know about then small set of phonetic units which act as input, and thus need not bother with analysis of text or any other input format.

This type of paradigm is commonly used in speech synthesis where a chain of modules map from an input language to an output language, which then becomes the input language of the next module in the chain. Having a phonemic level in a speech synthesizer is useful and powerful as it is there that the boundary between sound and purely linguistic aspects of language exists.

Although some older intonation systems used a single stage to map from (say) a syntax tree to an F0 contour, the common approach these days is to map from a syntactic/semantic representation of an utterance, to a phonological specification of the intonation, and via a phonetic component to a F0 contour (Pierrehumbert and Beckman, 1991), (Monaghan, 1991). Thus the problem now becomes that of deriving a phonological description of the intonation from the input, and then phonetically interpreting this description and producing an F0 contour.

A fundamental point where intonation differs from phonemic aspects of speech, is that there is little agreement of what the phonological intonation system for a language such as English is, and furthermore, what the phonological description of a given utterance should be.

Most modern phonologists and phoneticians have abandoned the belief that phonemes as traditionally defined are a true model of the phonology of languages. However, they are still very useful approximations to the true model, and provide a easy language with which to describe sound patterns. Furthermore, for a given a word in language, most linguists would quickly agree to an approximate description of what phonemes are contained in that word. This can be done by either listening to the word or from text. Intonation, on the other hand is much more troublesome.

Intonation in English is non-lexical which means that a given sentence or word can be spoken with many different acceptable intonation patterns. Thus it is impossible to predict from text what the intonation pattern of a word or sentence is. (It may be possible to predict one of many suitable patterns, or even list all the suitable patterns, but providing a single correct pattern for an utterance is not possible). When describing intonation, linguists typically listen to an utterance and then use an intonational description notation to record what has been said. In some ways this is like the procedure of phoneticians making narrow phonetic transcriptions of utterances. The aim of a narrow phonetic transcription is to make an accurate recording of what was said, such that another phonetician can reproduce what was said. Although much work has been carried out in constructing intonation notation systems (Palmer, 1922), (O'Connor and Arnold, 1973), (Crystal, 1969), (Halliday, 1967), (Pierrehumbert, 1980), it is clear to me that none of these systems have anywhere near the power and accuracy of a narrow or wide phonetic transcription. A number of points should make the last statement clear.

- Intonational notation systems are very difficult to learn and use. I certainly haven't mastered any of the systems mentioned previously, and from experience I know that many people who are familiar with prosody still find it difficult to come to grips with a notation system, such that they can label intonation accurately.

- Usually there is no justification given for why a particular notation system should be the way it is. Typical reasons are "linguists have been using this system for years" or "this new system correct a problem with a previous system", neither of which are solid accounts. Occasionally experimental evidence is provided, such as the experiments performed by Liberman and Pierrehumbert (1984). While these studies certainly highlight particular points, they usually only give evidence concerning a small area of the theory.

  Compare this situation to that of traditional phonemics. There we have a set of criteria for determining the phonemes of a language (a kind of meta-theory), and thus its easier to objectively state what phonemes exist. If our task was to design a intonation notation system for a different language, we would have to start from scratch and previous knowledge of English intonation would only be useful in a very indirect way. I am not advocating a return to a strict Structuralist approach, but a set of principles for saying what a intonational notations system should describe and how to go about designing one for a new language would be very useful.

- Although the previously mentioned systems agree on a significant number of points, there is also a substantial amount of disagreement. For example, none of these systems agree on how many types of pitch accent exist in English. In addition, many of these systems advocate the use of more basic intonational primitives, such as high and lows or rises and falls. There is much disagreement about what these primitives should be. Again, if we make a comparison to phonemic analyses, we also see disagreement there, but that is mainly constrained to the finer points, rather than disagreements about the fundamental units.

- The most convincing argument arises from the difficulty in reproducing an utterance's intonation. Take an utterance, transcribe it phonetically, give it to another phonetician and ask them to speak it. The resulting speech is usually recognizable as having the same phonemic content as the original.

With the equivalent test for intonation, things are much more difficult. Even experts in intonation have trouble reproducing intonational notations in this way, which points to the inadequacy of the notation system.

For many types of contours, we do have adequate notations, such as the "calling contour" and the "surprise/redundancy contours" of Liberman (Liberman, 1975). Such intonational patterns are easily reproducible, but unfortunately this taxonomy only covers a small range of the possible intonational effects.

With more traditional phonological systems, one often find that they are fine at dealing with simple intonational situations. For example, the O'Connor and Arnold system was designed partly to enable English language learners to improve their English intonation. In that sense, the patterns describe are those of a idealized fluent speaker, and the system restricts itself to those things which a language learner would choose to learn. It doesn't not cover many other intonational affects such as those due to sarcasm, shouting, contradiction, incredulity etc. In my view the perfect intonational description system should deal with such phenomena, as they are as much apart of day-to-day language as anything else. Intonational descriptions systems are more similar to say syntactic systems than phonemic systems, in the sense that they can handle typical clean utterances, but fail to give any account of the language that is naturally produced.

Thus a fundamental problem in designing an intonation module in a speech synthesizer is the lack of an adequate description language for either the phonetics or phonology of language. Because of this, and also because of the many different ways in which a set of words can be spoken intonationally, it is very difficult to describe the sound pattern of an utterance's intonation.

1.2 The "Discourse" Level

So the "words → phonemes → acoustics" chain is difficult to recreate for intonation due to a lack of an adequate description system for intonational sounds (which would replace the "phonemes" part). So what should the equivalent of words be? We have already said that a given sequence of words can be spoken with many different intonations, so it is unlikely that words are suitable. Unfortunately, there is no consensus here either. Linguists disagree as to to whether syntax, semantics, pragmatics, various combinations of these, or even non-linguistic factors are the contributing factor in deciding intonation.

In the current intonation synthesis module, we have a concept of the *discourse* level. An utterance transcribed with this information has syntax, speech act, focus, new/old information etc. It is from a discourse representation that we are trying to synthesize F0 contours, via the phonology.

1.3 The Issues

This report only concerns the intonational process between the phonology and the F0 contour. A common definition of the task is to devise a mapping between the phonological notation and the F0 contour. Here we extend the scope of the problem to include the design of the phonological notation system too. Thus goal of this part of the work is to produce a description language which can describe any contour we come across, and to provide an algorithm which can produce a F0 contour given this description. As we want our synthesis system to be as similar to a human voice as possible, we want our intonation component to be able to produce any f0 contour a human can.

The particular issues that are important include:

- Pitch accent description. How many types of pitch accent are there? Are there a fixed number? What F0 shapes do they have?

- How is downdrift to be modelled? (see section 9.3). Is it a property of phrases, pitch accents? Is the principle cause Downstep or declination?
- How does the segmental content of the utterance interact with the F0 contour? It is obvious that during unvoiced segments there is no F0, but it is also though that difference types of vowel, consonants and syllables also affect F0 shape.
- How is pitch range to be controlled?
- How does one "tune" the intonation module so as to mimic a different speaker's voice?
- How does one test a system?

2 The Approach

2.1 Knowledge Based Approaches

There are many general approaches to problems in speech synthesis, but a particular attitude can be referred to as the *knowledge based approach*. Maybe the classic example of this approach is the MITtalk system (Allen et al., 1987), principally developed by Denis Klatt at MIT. Unquestionably this system is of high quality and it is only more recently that other systems have caught up. In the phonetic component of MITtalk, Denis Klatt used his expert phonetic knowledge to hand craft a set of rules that when used with a formant synthesizer produced good quality speech. The problem with this approach is that the method with which Klatt designed these rules is not explicitly defined but was rather the accumulation of his years of experience. This is problematic in that if one desires to improve or expand the system, there is no well defined way of doing so. If we were told to make a model of a particular speaker's voice, we would not be much better off having MITtalk than beginning from scratch. Even more difficult would be to adapt the system to a new language. Such a task could only be done by having a expert phonetician of Klatt's ability for the new language.

In intonation this approach is often adopted. Many algorithms giving sometimes good intonation exist, but it is not clear how one would improve the system to model a new intonational effect, a new speaker or adapt the system to a new language. Often these algorithms are only tested in very ad-hoc ways, either by using informal listening tests, or by using more controlled perceptual evaluation. While perceptual evaluation is certainly useful, it is not a very powerful testing paradigm, as listeners may say that a certain technique is either more or less natural, but it is difficult to pin point the source of the unnaturalness. Again, during development of a system, it is impractical to conduct formal listening experiments at every stage.

2.2 Statistical Approaches

The other common approach is the statistical approach, whereby an algorithm learns the mapping between some abstract description of the intonation and the F0 contour. Such approaches are attractive in that the can be adapted to new data simply. It is also often easy to test these approaches and give figures for performance, which are very useful when comparing systems. However, if no constraints are used, a very large amount of training data may be required, and this defeats the goal of quick adaptation to new speakers and languages. Also, effects which are prevalent in the training data can swamp those which are less common, but which may be important in the language.

2.3 Synthesizing and Comparing to Real Data

My approach can be viewed as a sort of hybrid approach between the knowledge based approach and the statistical approach. Data is used in the development, training and testing of the model.

Using real data is a help in developing the model, one does not have to rely on ad-hoc anecdotal evidence to design the intonation algorithms, one can easily try and test new ideas by looking data.

With sufficient data and appropriate learning techniques, it is possible to train the parts of the model that need to be trained. Simulating the intonational characteristics of a particular speaker should only be a question of obtaining some labelled data from that speaker.

The analysis/re-synthesis technique is a powerful testing paradigm for this type of work. A database is used which has been labelled with the discourse and F0 levels. Any algorithm which purports to perform this mapping can be tested by giving it the discourse level as input and comparing its F0 output with that of the original. The perfect intonation module would be capable of mapping from the discourse level to the F0 level with complete accuracy for any utterance. Note that using real data for testing is still valid regardless of whether the synthesis algorithms are statistical, rule-based or other.

2.4 Designing New Levels

Many new levels are proposed in the work presented here, and by using the above paradigm, it has been possible to test how useful these levels are in helping in the overall problem. At the discourse level, we can expect information such as syntax, speech act and focus information etc. to be present. Thus the design of the phonological level, or any other level, is a compromise between describing a level that is producible from the discourse information and one that can be interpreted to produce F0 contours. In other words a description system that is easy synthesize *to*, and one that is easy to synthesize *from*.

One can argue that in the "real" model of intonation production, the mapping between phonology and acoustics is done in a single stage (Pierrehumbert, 1980), in practice, and especially in computer implementations, the practice of splitting a larger problem down into several smaller problems is an accepted method of procedure. In the intonation system described here, there are many levels used between the highest and lowest levels. Working with these levels has made the overall problem easier to manage, but we are not at the stage where we can say whether these levels have any place in the "reality" of intonation.

Testing mappings is performed by taking labelled input from real utterances, passing this trough the mapping and comparing the output to the what is labelled in the real utterances. In this way one can design and improve the mappings. As the output of one mapping is the input to the next in the chain, it is possible to combine a series of mappings into a larger one, and assess the overall performance in the same way. This method of operation should become clearer with the specific examples shown.

2.5 Data

Many intonation experiments and theories use very carefully spoken utterances, elicited to produce a particular effect. I don't want to use this approach for several reasons. Firstly, the experimenter runs the risk of biasing the experiment. Secondly, if one were to design a separate experiment/data set for every known effect in intonation, it would take for ever. Thirdly, we want our system to be able to speak the way real people speak, and not according to how traditional linguistics thinks they should speak.

All the experiments conducted here used natural data that was recorded for purposes of having a general database for speech recognition. This data comprised of acted conversations in the conference registration domain. This was the test domain for the ATR spoken machine translation system, and so was particularly suitable for the ATR speech synthesizer. The data is not fully spontaneous, as there are

no hesitations or re-starts, and because the speakers were told in advance what to say (but not how to say it). However, the data did contain a variety of types of speech act, intonational tunes, phrasing, focusing etc and the utterances were from a number of American speakers.

Although more spontaneous data is always a tougher test, I think this particular data set is ideal for speech synthesis purposes as it is fluent (we don't necessarily want stutters in out synthetic speech), but also natural and expressive. In short, this data is the type of speech we would like our synthesizer of being able to produce in a machine translation system.

3 Summary of Previous Work

Before coming to ATR, I had spent about 18 months working on an intonation model which is called the RFC model (Taylor, 1992). The thesis describes more thoroughly many of the issues concerned with modelling intonation. Two new levels of intonational description were defined; the RFC (rise/fall/connection) level; and the HLCB (high/low/connection/boundary) level. The mapping between the RFC level and the F0 contour was defined using a set of equations and an automatic labelling system was devised which attempted to produce an RFC description given an F0 contour. The HLCB was a phonological description system, but only vague indications were given as to how to relate this to the RFC level.

The work described here builds on the solid foundations of the RFC system, but the HLCB system has more or less been abandoned. The HLCB system suffered from the same problems that all the other phonological system suffer from, i.e. arbitrary reasons for it being like it is. I could give no evidence for why it was a good system, and the arguments I put forth for it being better than, say, the Pierrehumbert system were esoteric at best (although maybe true). There was also the problem that I could not hand label with the HLCB system with much more reliability than when using any other system. Furthermore, the design of the desired mapping between it and the RFC mapping was not proving easy. For these reason, this system was abandoned and work was channelled into designing a new phonological system starting from the RFC level and working up. Designing a phonology from a purely bottom up perspective is maybe impossible and unwise as the phonology should be the link between the discourse and F0 level. So the broader goal is really to work in both directions, from the acoustics up and from the discourse level down, and in that way try to negotiate a new phonology and series of mappings.

In contrast, the RFC system has proved its worth. It had been designed using only Irish and Southern-English utterances described in my thesis, and so there was some doubt as to whether it would expand to different dialects. In experiments carried out for my submitted paper to Speech Communication (reproduced in the appendix), it was actually the case that the RFC system worked *better* on the American CMU data than on the original data. When the utterances were hand labelled and re-synthesized, the differences between the original and re-synthesizer versions were smaller for the CMU data than the data in my thesis. However it was seen that the automatic RFC labeller, could only achieve about 75% accuracy on this data, so it was clear that much more work need to done in this area in the future.

4 A Model based on Intonational Events

The starting point for the new model is really just a re-phrasing of the previous RFC model. The new model is called the *event* model. The starting point as described here, is a classification system for the low level aspects of intonation, is a basic low level phonetic way of describing F0 contours. The "axioms" of the new model can be stated as follows.

- Speech can be divided into linear sequences of *intonation contours* which are delimited by silence. These contours can be viewed as "phonetic phrases" such that they need not necessarily be linked

to any syntactic or idealized prosodic structure.

- Each phonetic phrase has a single pitch range. This pitch range is effectively given by a single f0 value which states at what F0 value the contour is to begin.

- Within these contours there exist intonation *events*. Events are the real substance of intonation, and each event is linked to a specific act by the speaker. Pitch accents are the most important events in terms of their diversity of classification, but the other two commonly used events are question and continuation rises, and what can be referred to as *resets*, which are the usually small rises observed at phrase beginnings.

- Events are associated with syllables. Syllables may only be associated with one pitch accent, but sometimes an accented syllable can be associated with a boundary event also.

- All events have the same basic shape, which is a rise+fall pattern in the RFC system. Each event has 5 numbers describing its shape: the rise and fall have an amplitude and a duration, and an additional number, *peak position* describes the position of the rise/fall boundary (the "middle" of the event) from the start of the vowel of the associated syllable.

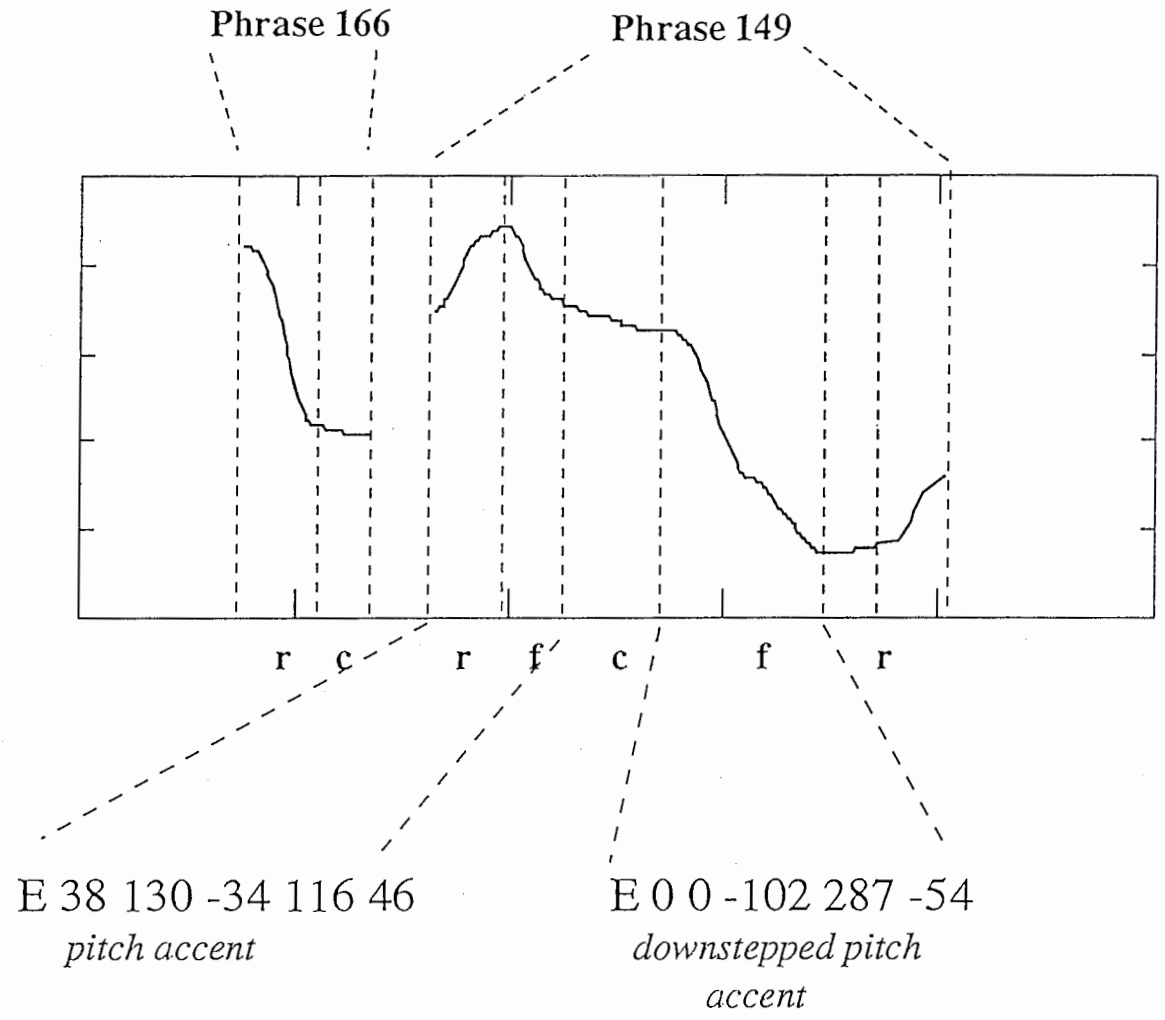
- Between every event there is a connection element, which is a straight line of variable gradient. The connection elements can be thought of as "white space" between the events; not very important in themselves, but necessary to complete the F0 contour. Sometimes events are very close together and in these cases the duration of the connection element is 0.

The most unusual aspect of this model is the impression that there is only one type of intonational phenomena. This is not really the intention, rather the model makes it clear that all intonational phenomena can be described using the same formal language. The differences in the 5 numerical parameters needed to specify the events distinguish the various types of pitch accent from one another and boundary rises etc.

It is possible (and very common) for either the rise or fall part of the event to be zero, such that the event only appears as a rise or a fall. In the analysed data, there is about a 3 way split between events which have only rise, those which have only falls and those which have both.

In the American data, there were many small rise-only elements which were not specifically linked to stressed syllables. These often served to raise the intonation at the start of a phrase, or raise the intonation at the start of a compound noun construction (which would usually later end in a fall-only accent). This later type of event, in conjunction with a subsequent fall-only event, is similar to the "flat-hat" accents of the Dutch school of intonation (t'Hart and Cohen, 1973). One might then propose that the rise event, the fall event and the intervening connection element are really a single event. This is certainly a possibility, but this proposal was not accepted as there seemed to be no restriction on how far before the fall event the rise event could occur. Furthermore, as the rise event becomes earlier it seems to become less like the rise part of a flat-hat and more like an unrelated reset. However, this issue maybe isn't so much of a problem as we know that *formally* the F0 has been described accurately, and maybe a further higher level phenomena would map a single phonological flat-hat to two separate events. This issue obviously deserves more investigation, but as far as I can see, this has not been a serious problem for the synthesizer so far.

Another, much more radical proposal is that there is no basic division between High and Low accents (H* and L* in the Pierrehumbert system). The proposal here is that there is only really one type of pitch accent, which is very similar to the types of H* accent. L*+H accents were controversial in the Pierrehumbert theory anyway, with many (eg. (Ladd, 1983)) thinking they were really H* type accents. Often, what are referred to as L* accents have no observable effect on the F0 contour near the accented



KEY: E 38 130 -34 116 46

rise ampitude *fall amplitude* *peak position*

rise duration *fall duration*

Figure 1: *An utterance's F0 contour and an event based analysis.*

syllable. They are classed as L* as there is the perception of a nuclear syllable be present, and because the contour rises at the end of the phrase. In the event model, this type of accent is ignored, at least at the phonetic level. The explanation is that there is an underlying stress structure to the utterance which provides a framework in which intonational events operate. Nuclear stress is signaled just as much by the stress structure as by F0 movements, and thus this type of L* is really a nuclear syllable with no pitch movement and a subsequently rising contour. The most controversial proposal is that concerning the H+L* type of accents, which were described as L [+antecedent fall] in the HLCB system. As mentioned in my thesis, these accents appear phonetically very similar to downstepped H* accents. From analysis of the American data, there doesn't seem to be any noticeable difference here, either in phonetic behaviour or meaning. The only thing which distinguishes them is that after the L* type, the contour rises. I therefore propose that these types be combined and that the job of signalling a L* accent falls to the connection element following the event.

4.1 From RFC to Event

The event system described above was termed the numerical-RFC level, as events are specified numerically (as opposed to by features) and are in the RFC-space (as opposed to the tilt space (see below)).

A simple program was written that could take an RFC description of the contour, a description containing syllables and durations and event locations, and produce a list of syllables which had events and their parameters marked.

The overall goal as stated earlier is to provide a mapping between the discourse level and the event level. In this task, the event based description of intonation is useful for a number of reasons.

- It provides a explicit formal way of describing intonation on a low level.
- Events are representative of underlying phonological events such that there is a simple relationship (usually one-to-one) between the events on the numerical-RFC level and the phonological level.
- F0 contours are continuous plots of fundamental frequency against time, whereas descriptions on the discourse and phonological level are usually considered to be discrete. The numerical-RFC event level has discrete units in a similar way to the phonological level, with the continuous parameters representing the variability between types of event.
- Events are described with respect to syllables, which is the standard way of describing pitch accents. Boundary and continuation rises are usually thought of as being associated with phrase boundaries, but associating them with phrase-final syllables should be clear. Associating reset events with syllables is maybe more questionable as I haven't really found any criteria for where they should be placed. By describing events with respect to syllables instead of segments or in isolation (as with the original RFC descriptions), we have made a much stronger link to the phonological level than would otherwise be.

4.2 Extending the Numerical-RFC Event Level

It is possible to synthesize F0 contours from the numerical-RFC event level with the same accuracy as from the old RFC level[1]. In the synthesis evaluation tests, it was shown that the F0 contours synthesized from hand-labelled RFC descriptions of utterances, were on average 4Hz different from the original F0 contours of those utterances. The importance of such results is discussed in the Speech Communication paper in the appendix, but it is enough to say here that this indicates a very high level of synthesis ability.

[1] Accuracy figures for this are given in the appendix.

Using events instead of the simple RFC descriptions brings the classification language of the pitch contours closer to the higher level descriptions, and thus it should be easier to generate event descriptions than original RFC descriptions.

However, the numerical event level is non-ideal. Firstly, there is the question of redundancy. Is it really necessary for events to have 5 parameters? (rise amplitude, rise duration, fall amplitude, fall duration and peak position). And do these 5 parameters represent the linguistic variation among events naturally? Obviously, the fewer parameters we have to generate from, the easier that generation will be, and furthermore, the more appropriate these parameters are to higher level descriptions the easier the generation will be.

A basic assumption is that information in the higher levels of the intonation system has less redundancy than that on lower levels. Thus it is much easier to predict an F0 value given the previous 5 values, than to predict what form a pitch accent will take given the previous 5 pitch accents. This is a standard assumption in phonology, where the lexicon is assumed to contain high-information content specifications, which are turned into more redundant surface forms during speech production. We don't yet know what our phonological description will look like, and so we don't know how to optimally change the RFC parameters so as to best represent this level. However, we can use the concept of redundancy to help us, as we know that a redundancy reducing mechanism should be a step in the right direction.

Redundancy can be assessed by measuring the correlation between the parameters. If we found for instance, that rise amplitude was proportional to rise duration, we could code these values in a single variable, and thus reduce the number of parameter from 5 to 4.

Two methods were tried in order to reduce the redundancy and produce events with fewer parameters.

5 Principle Component Analysis

Principle Component Analysis is a standard technique for finding the optimal independent axes of description for a series of observations. In our case, we have 5 dimensions corresponding to the 5 numerical-RFC parameters, and we would like fewer, say 4.

Principle component analysis finds a new coordinate system for the data, such that the first axis has maximal variance, and the fifth axis has minimal variance, and, that all the new axes are orthogonal. Thus it maps the original data into a new space where the axes vary independently. It is simple to perform the inverse mapping and transform an observation in the new space back into the original space.

The usefulness of principle component analysis lies in the fact that the axes are now independent and ranked in order of variance. To reduce the number of variables to 4, one just ignores the variable with the smallest variance. Obviously, there is still some information being encoded in this variable, so it is useful to see how bad an error is incurred by having this reduction. To assess the error, the mapping of the original matrix is performed, the values in the smallest varying variable are set to zero, and the inverse mapping is performed. The resultant matrix is then compared with the original using a comparison criteria. As mentioned previously, the choice and design of a comparison criteria is a non-trivial problem.

The numerical-RFC event level has 5 parameters, 2 of which are measured in Hertz and 3 of which are measured in milli-seconds. They all have different means and variances and so it would seem practical to normalise these variables before comparison. Simple statistical normalisation makes each variable equally important, but this may still not be ideal as we don't know if they really are equally important from a perceptual point of view. No effort has yet been made to produce a more sophisticated comparison criteria beyond normalisation, but this area is obviously a suitable topic for further work. The error criteria that was used gave a score of the absolute distance between each variable in the original data to each variable in the reconstructed data. By measuring this over all observations it was possible to obtain a score for the general accuracy of this application of the principle component technique.

To give an impression of the maximum error, a special matrix was constructed that had 0 values for all variables for all observations. The differences between each reconstructed variable and the original were divided by the maximum error for that variable and multiplied by 100 to give a percentage score for each variable. By averaging the five percentage scores, a single score for that technique was found. The results are given in table 6.

5.1 Problems with Principle Component Analysis

There are a number of advantages and disadvantages to using principle component analysis. It is advantageous because the technique is well defined and mathematically guaranteed to give an optimal solution. Also it need no knowledge of the what the axes actually represent.However, there are also significant disadvantages. Firstly, the technique is heavily dependent on the data used. It would be extremely unlikely that a principle component analysis of a different data set would produce the same mapping matrix. More importantly, the new space that is mapped into, would be different for different datasets, i.e. different speakers. One would hope as we move from the acoustics to the phonology that the descriptions would become less not more dependent on a speaker's characteristics. Finally, the power of the mathematics in standard principle component analysis is limited to first order operations, i.e. the values in the new space are calculated by a matrix multiplication of the mapping matrix on the values in the old space. It is possible that more powerful mathematical techniques could produce better results.

6 Tilt Theory

A "knowledge-based" alternative to the principle component analysis was designed and proved of sufficient worth to be adopted in the synthesizer. The knowledge was based on a few informal observations concerning this data and others.

- In the numerical-RFC level, there are two variables representing amplitude. In many rise-fall events, it seems as if the sizes of the rise and fall are strongly correlated. Thus it natural to code these two values with a single variable.
- The above observation is also true of duration.
- The seems to be a natural way of classifying the "orientation" of events, ranging from rise-only, to rise-fall events, to fall only.

The *numerical-tilt* level is thus comprised of *amplitude* which is the sum of the absolute amplitudes of the rise and fall component; *duration* which is the sum of the duration of the rise and fall component; *peak position* which remains as before; and *tilt* which is a measure of the orientation of the event.

The first three variables are straightforward enough, but it is the concept of tilt which deserves the most discussion. Much investigation was carried out in specifying exactly how to describe this. It was decided that it would be desirable to have tilt have dimensionless normalised values. For example tilt could be derived from rise and fall amplitude alone, and described as the difference of the two amplitudes divided by the sum. The basic tilt equation is given in 1 or more generally as in 2.

$$tilt = \frac{|A_{rise}|-|A_{fall}|}{|A_{rise}|+|A_{fall}|} \quad (1)$$

$$tilt = \frac{A_{difference}}{A_{sum}} \quad (2)$$

Method	rise amp	rise dur	fall amp	fall dur	peak	total
Princ. Comp	43.0	2.3	0.4	0.0	0.1	9.2
Normal Princ. Comp	23.9	3.5	22.3	1.2	0.6	10.3
Linear Tilt	10.0	5.0	7.0	4.0	0.0	5.2
Log Tilt	12.0	6.0	3.0	2.0	0.0	4.6

Table 1: *Results in Percentage Error from Principle Component and Tilt Analysis. The first row gives the percentage errors for standard principle component analysis. The second row gives the results when the input is normalised before analysis. The third row is the simple tilt scheme and the last row is the tilt scheme where the amplitudes have log instead of linear values.*

$$A_{sum} = A_{rise} + |A_{fall}| \tag{3}$$

$$A_{diff} = A_{rise} - |A_{fall}| \tag{4}$$

Rise-only events have a tilt of +1, rise-fall events with equal rise and fall amplitudes have a tilt of 0, and fall-only events have a tilt of -1.

The inverse mapping requires the production of rise and fall amplitudes from the new amplitude and tilt values[2].

$$A_{rise} = \frac{A_{sum}(1+tilt)}{2} \tag{5}$$

$$A_{fall} = \frac{A_{sum}(1-tilt)}{2} \tag{6}$$

Tilt can also be calculated for duration using formula 1 but substituting duration for amplitude. It should be clear that with the 5 values of amplitude, duration, peak position, amplitude tilt and duration tilt, it is possible to map from the RFC space to the tilt space and back with no error. The real usefulness of the tilt system lies in the very string correlation between the amplitude and duration tilt, which allows these to be combined into a single quantity.

Figure 2 shows the 4 tilt variables and shows 5 events with different tilt values. Figure 3 shows amplitude tilt plotted against duration tilt for 190 events from speaker MAEM. A significant number of points are at (1,1) and (-1,-1) which correspond to rise-only and fall-only events respectively. It is clear that there is a strong positive trend in the data, and that there are no bad outliers to this trend. Figure 4 shows tilt values calculated for log amplitude against the same duration tilt values as before. Here the data is even more ordered. It would be possible to try many different ways of scaling the tilt variables so that the correlation increased, but here the log amplitude/linear duration combination was used. Although it would obviously be desirable to investigate the relationship between amplitude and duration tilt more closely, it was clear that the chosen technique was very successful and so therefore work was directed at building upon the tilt level rather than perfecting it.

Table 6 shows the individual and combined errors of the principle component and tilt theory analysis.

It is clear that the tilt system works at least as well as the principle component analysis. With more data pre-processing it may be possible to improve the principle component scores. An interesting point concerning the tilt errors is that they are more evenly spread than the principle component analysis which tends to concentrate the errors in the rise amplitude or duration.

[2]Fall amplitudes are normally represented with negative values, and it is important therefore to make sure this is accounted for when calculating tilt. It is possible that a more elegant formulation that explicitly models this might be better.

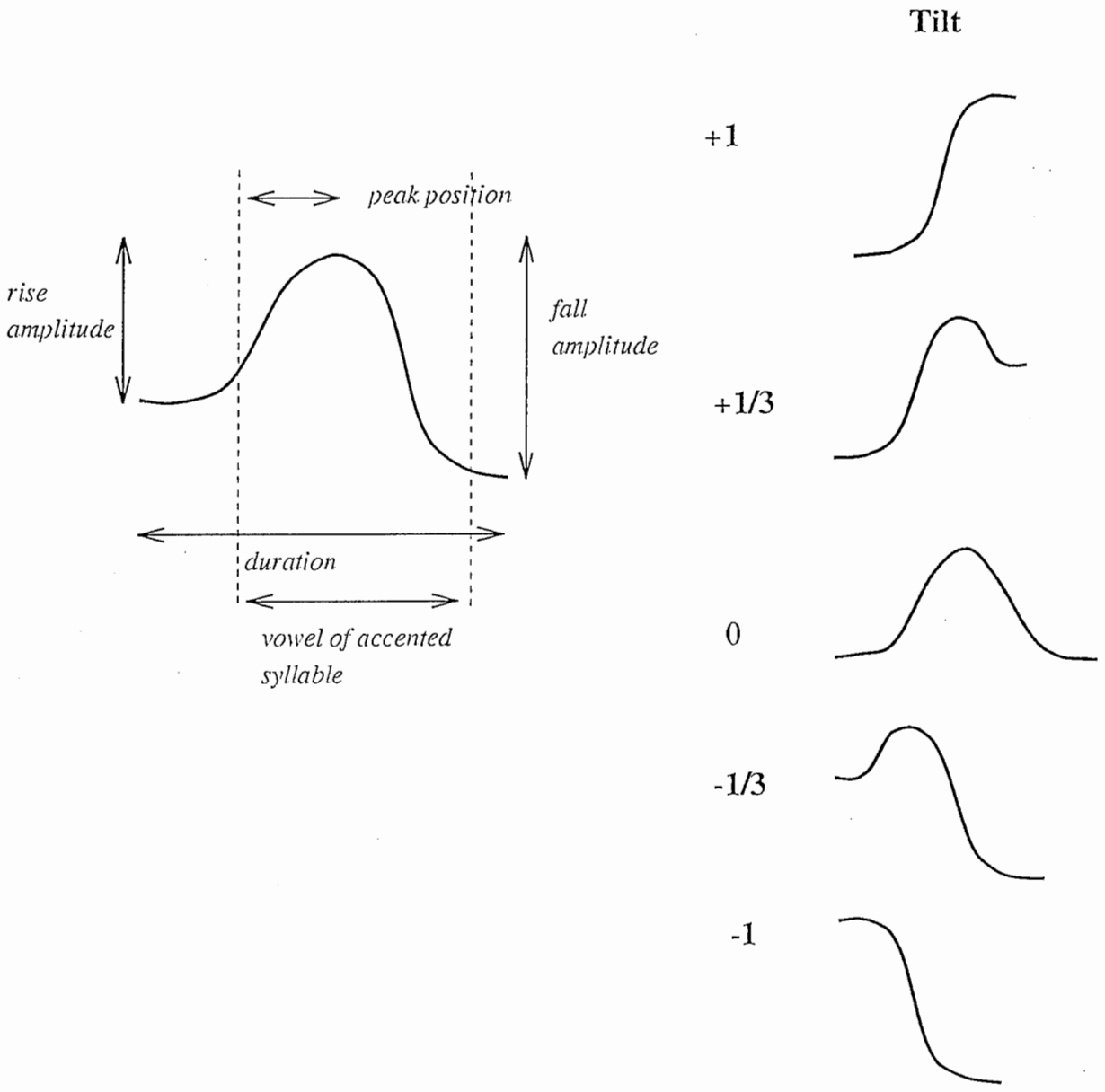


Figure 2: *The diagram on the left shows how amplitude, peak position and duration are represented in an event. The diagrams on the left show 5 different values for tilt.*

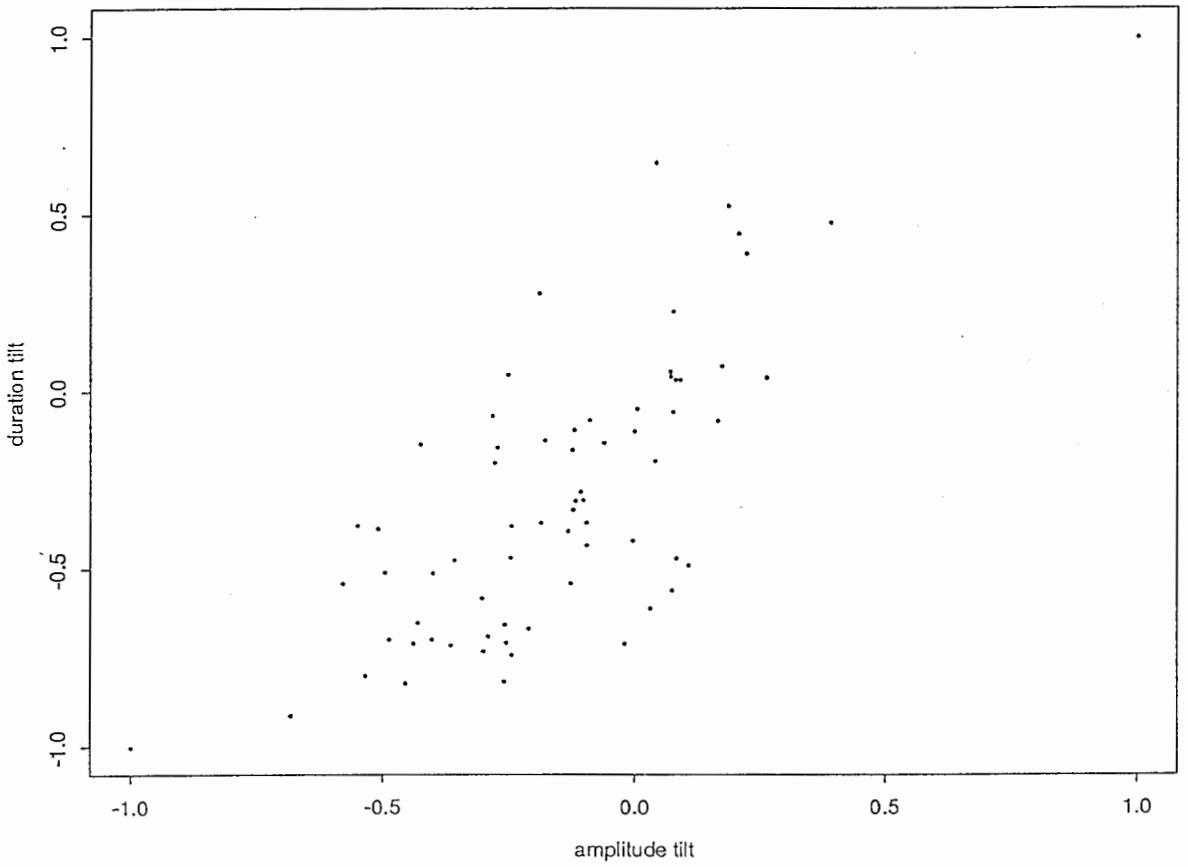


Figure 3: *Amplitude tilt vs. duration tilt.*

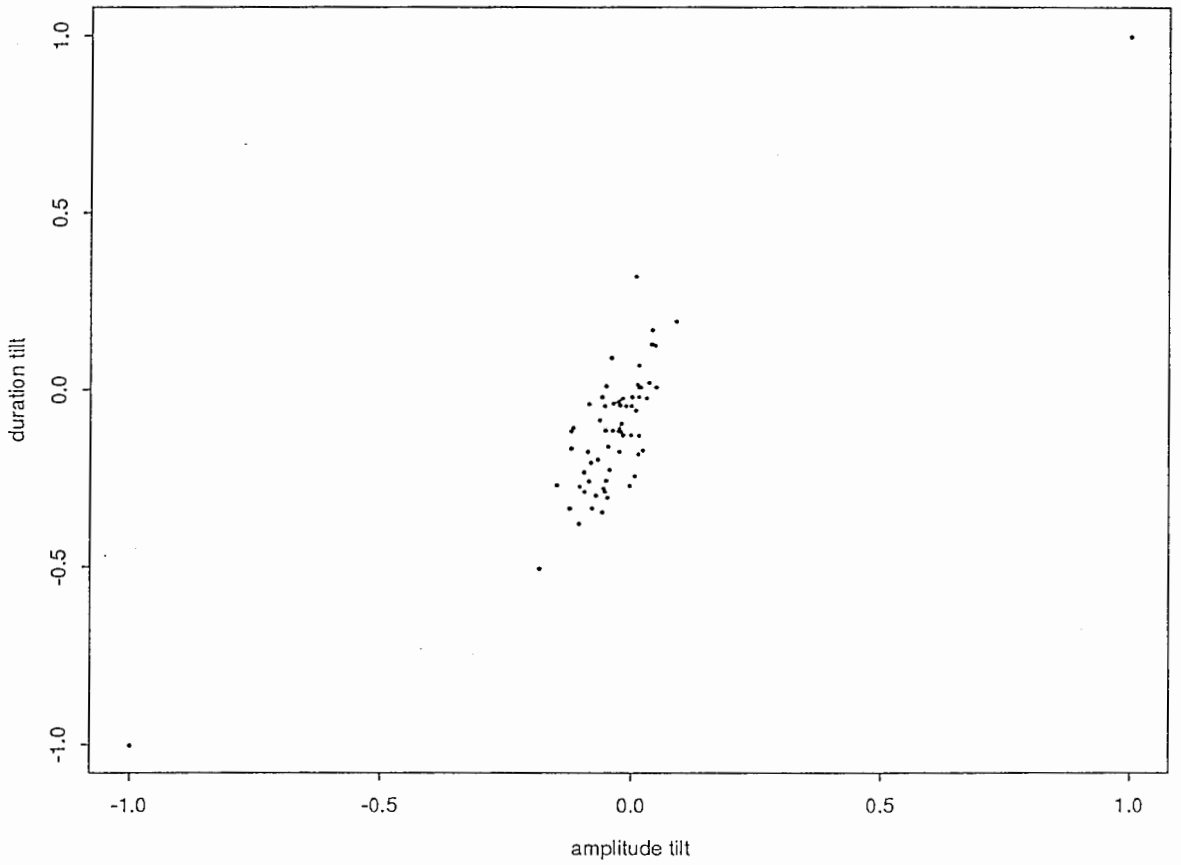


Figure 4: *Log amplitude tilt vs. linear duration tilt.*

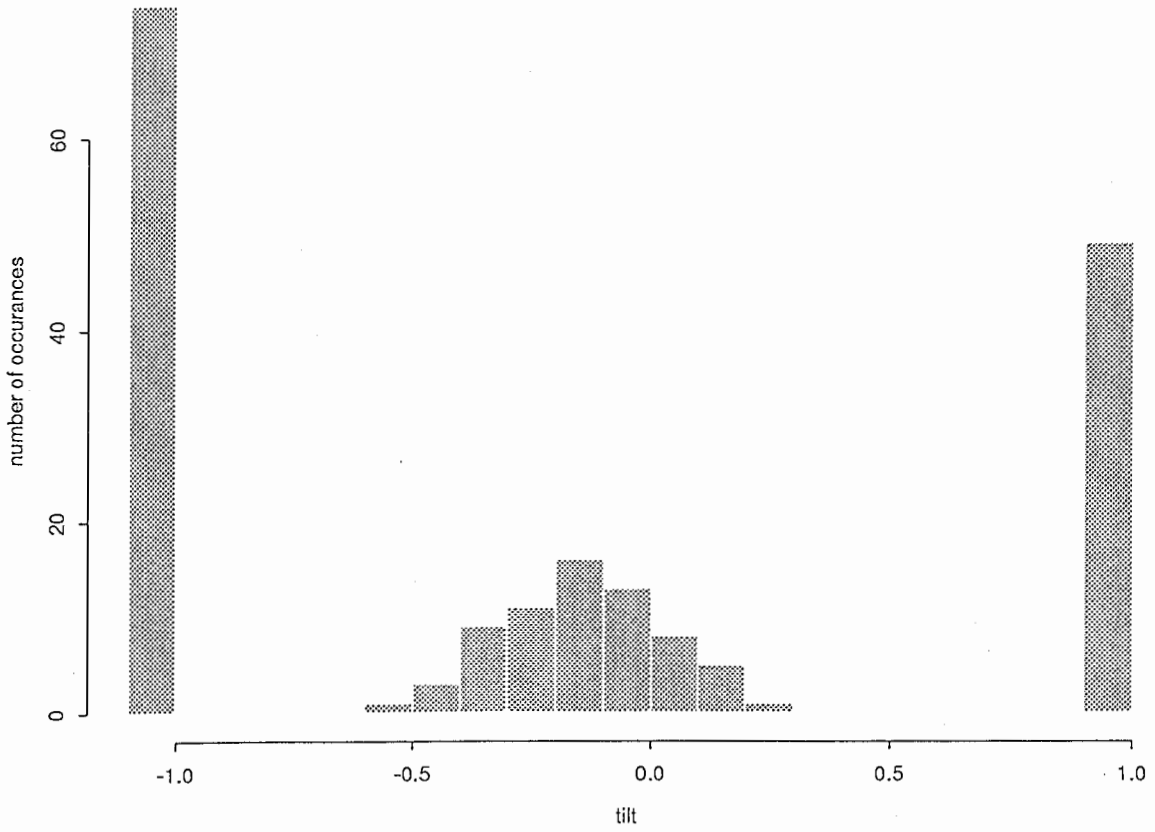


Figure 5: *Distribution of tilt for 190 events from speaker MAEM. There is a clear tri-modal distribution here.*

However, the main advantage the tilt system has over the PCA is that the mapping and new space are speaker independent in that the same equations and the same variables exist for all speakers. Also, the variables in tilt space are easily interpretable unlike the principle component variables.

7 Features

A lot of variance is seen when the distributions of the tilt parameters for a speaker are plotted. For 190 events from speaker MAEM, figure 5 shows the distribution of tilt, figure 6 shows the distribution of amplitude, figure 7 shows the distribution of duration, and figure 8 shows the distribution of peak positions. Why is this variance present? The variance can arise from two factors, the phonology or from the mapping which converts this phonology into a tilt description. How the phonological and phonetic factors give rise to the variance will now be discussed.

In the simplest case, one could produce a one to one mapping between each event on the tilt level and each event on the new phonological level. This would be done by proposing a mapping such that a tilt value of so-and-so combined with a particular peak position would map to a discrete accent type. There are a number of factors that prevent such a mapping being plausible.

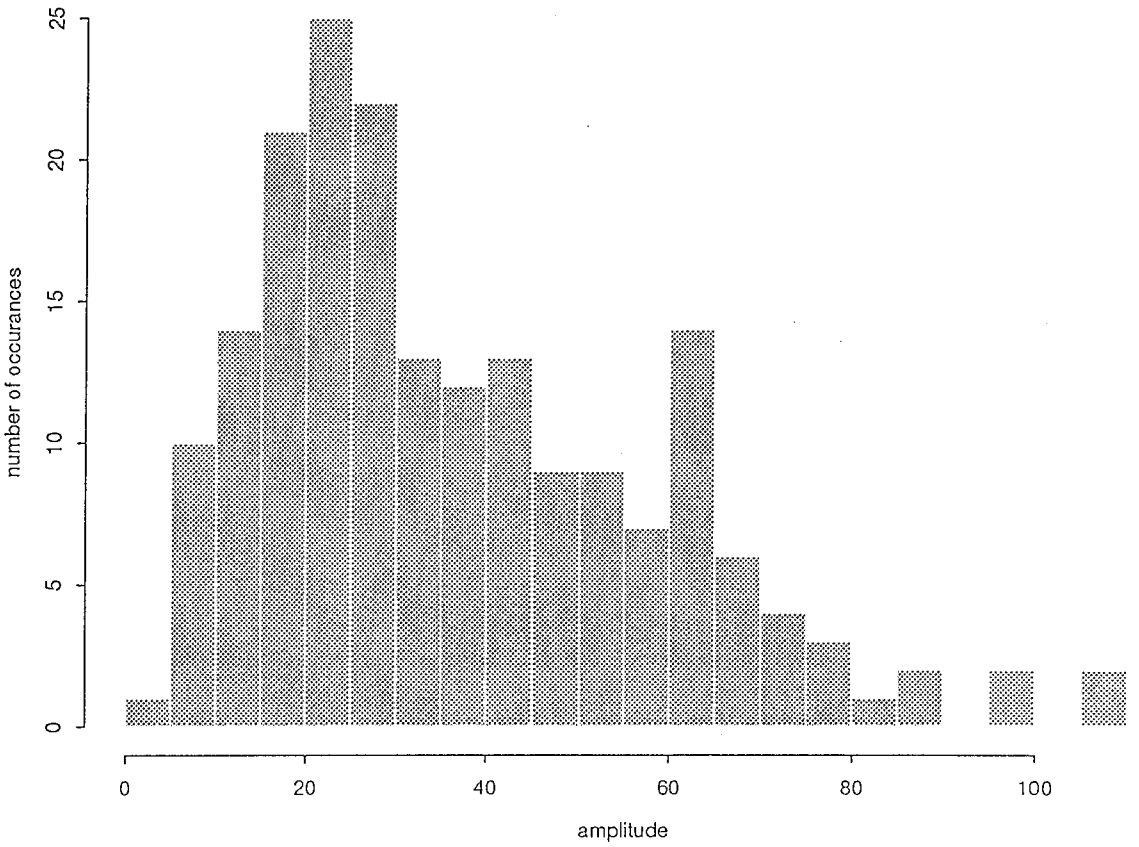


Figure 6: *Distribtion of amplitude for 190 events from speaker MAEM.*

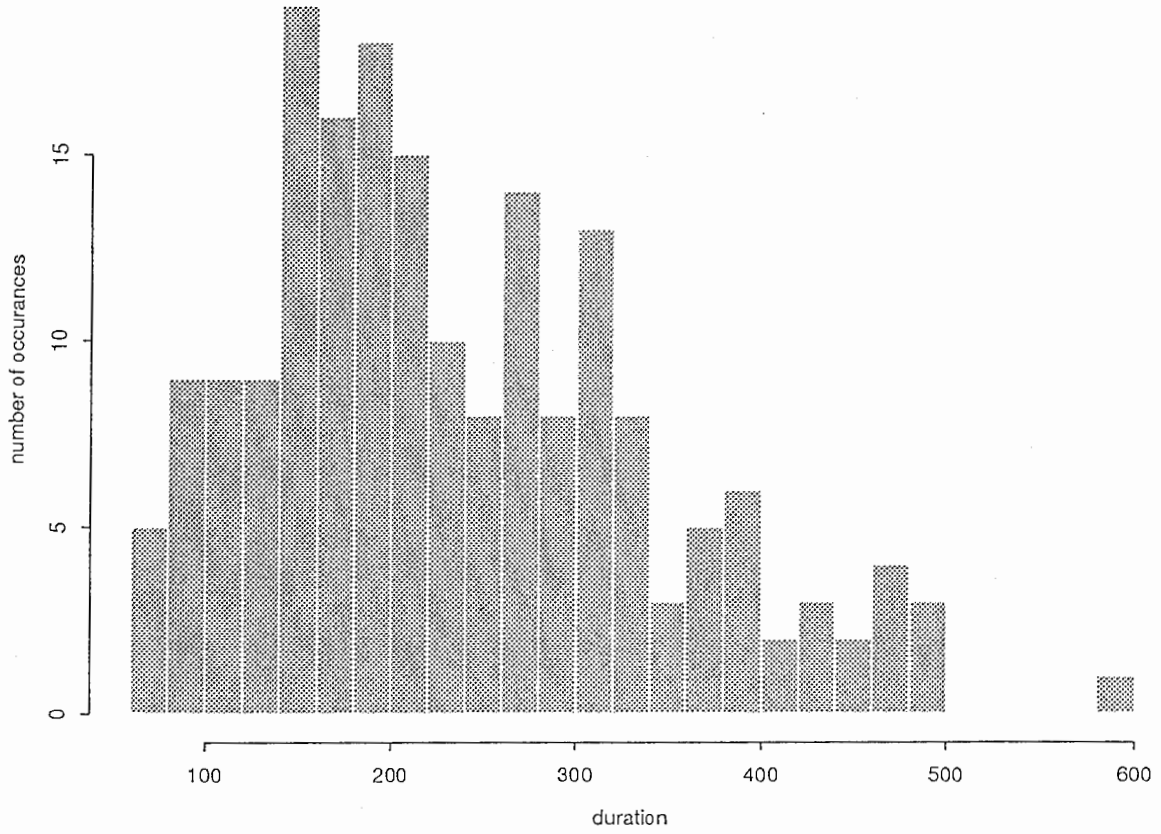


Figure 7: *Distribution of duration for 190 events from speaker MAEM.*

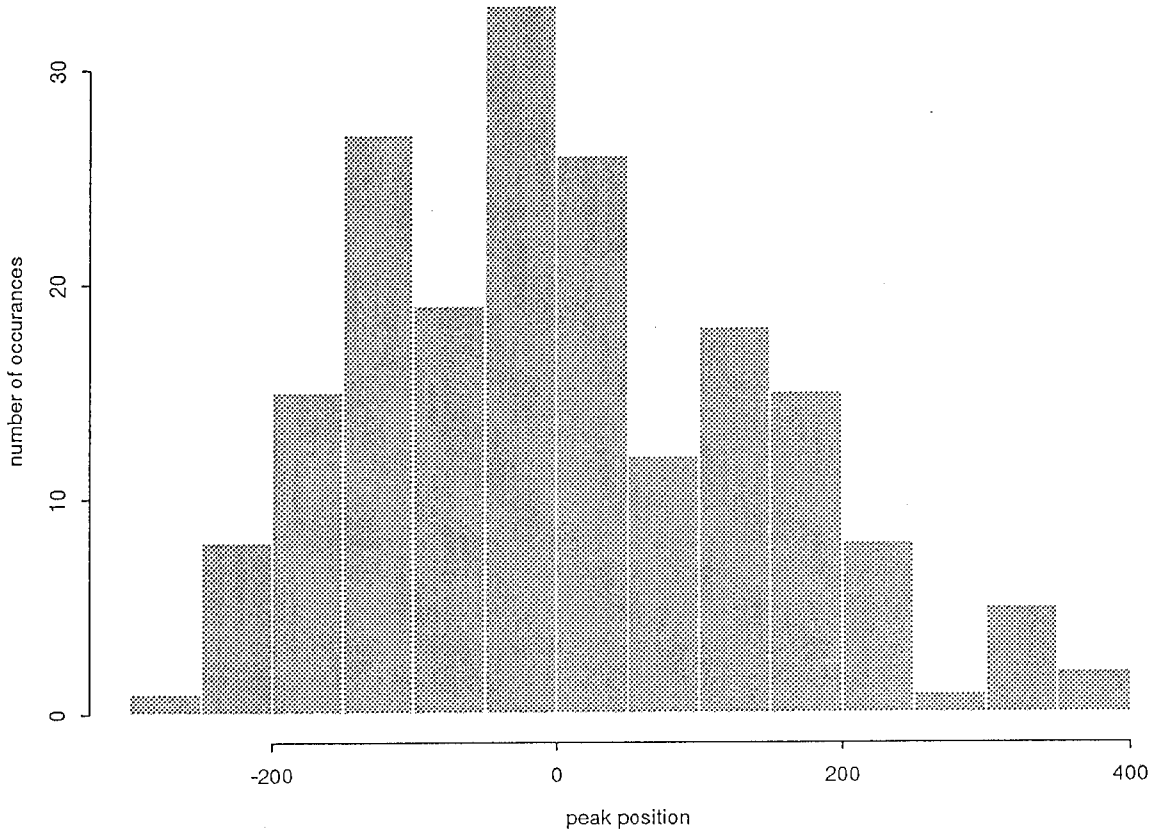


Figure 8: *Distribution of peak position for 190 events from speaker MAEM.*

The values in a tilt description are not speaker independent. At the very least, one would expect speakers to have different pitch ranges, especially if they are a different sex. Also, we have so far assumed a sort of "context-free" system, whereby events are not influenced by neighboring events. It is evident from the data and supported in the literature that this is not the case and that events will be effected by a number of factors. The following factors are thought to effect the behaviour of events.

- Phonetic effect of other events. Sometimes two events occur such that they overlap and when they do, both events are affected. I think the basic pattern is that the resultant two events are the "greater of" of the two underlying events, i.e., the amplitude, duration and peak position of the originals remain unchanged, but only the tilt changes.

- Phonological effect of other events. The well known stress shifting phenomena is relevant here, whereby stress if shifted from a syllable to a neighboring syllable so as to avoid a stress clash (two neighboring stressed syllables). Thus the events don't change in themselves, but the surface event is associated to a different syllable than the underlying stressed syllable. This happens a number of times in the data, for instance with the word "hotel" where the stress often occurs on the first syllable.

- Phrasal effects on position. It is somewhat obvious that events must occur within phrases, and it appears that events at the beginning of phrases are moved to the right and those at phrase ends are moved to the left, thus ensuring in both cases that the event occurs within the phrase.

- Phrasal effect on amplitude. It is often stated that the width of the pitch register narrows as a phrase progress, which would suggest that events at the end of phrases would have smaller amplitudes that those at the beginning.

- Segmental and syllable structure. It is also often stated that the types of segments which make up the event syllable will affect the shape and timing of the event.

Thus it is unlikely that the phonological description of an event can be derived without taking the above mentioned effects into account. The question is, how best to proceed?

The normal formulation of this problem would be something like: A discrete phonological specification for a phrase is passed through a phonetic realisation module which produces a surface phonetic description. The phonetic realisation module deals with the the above mentioned effects.

In a indirect way, all the phonetic realisation factors are present and known. i.e. it is possible to describe an event, its neighbouring events, the syllable it is associated with, the phrasing situation etc, as all these factors are present in the tilt-event description of an utterance. If the underlying phonological specification for each event were known, then one could construct a mapping that took the phonological description as input, modified it according to the phonetic realisation principles, and produced an tilt event level description as output. This is more or less the way traditional phonological grammars such as SPE (Chomsky and Halle, 1968) work. But, unless we arbitrarily adopt one of the discredited intonational notation systems described in the introduction, we don't know what the phonological specification for an utterance is (even if we did adopt one of these schemes, we still have the problem of knowing what description particular utterance should have). In fact our problem is remarkably similar to the initial problem of SPE. In that work Chomsky and Halle have many examples of surface level forms of words (from a dictionary) but did not know *a priori* what the underlying form was. Thus they had to deduce the underlying phonological description *and* the process of mapping between this underling description and the surface form.

Imagine the case where the phonetic realisation component makes trivial modifications to the underlying description, such that the surface and underlying levels can be considered equivalent. The variation

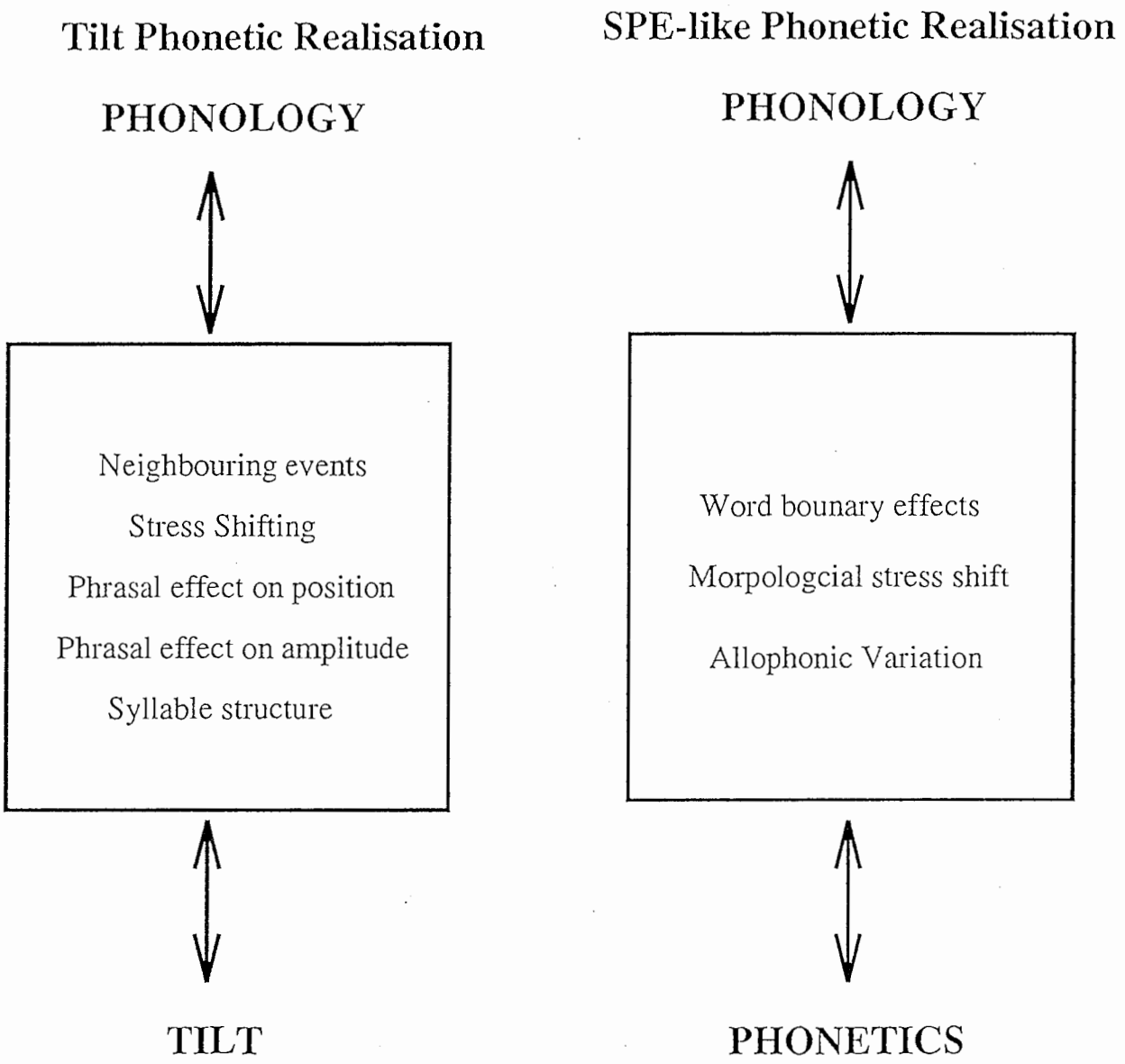


Figure 9: *The analogy between standard and intonational phonetic realisation in the tilt system.*

in the four tilt variables for pitch accent events would be purely a property of the differences between phonological pitch accent classes. It might then be the case that the events would naturally cluster into the phonological classes. (Say there were 6 pitch accent classes, then in the four dimensional space mapped out by the tilt variables, there would be 6 clusters formed by the pitch accent events in our data, one for each pitch accent class). It is frequently noted in traditional phonology that two underlying forms can give rise to the same surface form, thus making it impossible to study the surface form and deduce the underlying form. This may indeed happen in the intonation case, but the effect must be limited somehow as it is the case that listeners need to be able to deduce the underlying forms from the surface forms, and if too much ambiguity exists, then this will be impossible. Thus there are two possibilities, complete separation of classes based on tilt variables, or partial overlap limited by the need to disambiguate. (The disambiguation of pitch accents, boundary rises and resets does not appear to be a problem: pitch accents always have a fall, boundary rises only occur phrase-finally, and resets are the remaining isolated rises.)

Also of great significance, is the debate about whether underlying pitch accent specification can really be described in purely discrete terms. Pierrehumbert, for example, has always maintained that pitch accent prominence is a "para-linguistic" effect, which has the implication that prominence can only be defined in scalar terms. Thus, *a priori* we don't know how much of the underlying description is discrete and how much is scalar, with either of the two extremes (totally discrete or totally scalar) being possibilities. In the case of a underlying scalar variable (which may or may not be parallel to one of the tilt dimensions) we could still derive this from the shape events formed in the underlying tilt space.

The basic model is therefore that underlying phonological accents are specified in terms of a fixed set of discrete classes each of which has a set value for each tilt value. Scalar values may also exist which control one or more of the tilt dimensions. This is then fed into the phonetic realisation module which modifies the values of the four tilt variables according to the factors listed above. The resultant surface events then show all the variance and continuity observed in our data.

Given this situation, there are two possible paths. The first, most probable approach, would be to design a reversible phonetic realisation module that could take the events in our data and derive the underlying forms. Experiments showing how event variables vary with respect to syllable structure, phrasal position etc, would help in the design of this module. The second approach is too assume that the phonetic realisation module doesn't actually make that much difference, and that the surface and underlying forms are in fact quite similar. In that case we could immediately attempt a partitioning of the four dimensional tilt space into phonological classes.

The latter approach was the only one yet attempted. I tried this approach first, because it would be simpler to carry out, even though I was fairly sure that the phonetic realisation module is not trivial. As it turned out, I think the first approach is the correct one, but taking the second approach was still of benefit.

7.1 Labelling Features

My paper presented for the prosody workshop in Lund (Taylor, 1993) discussed a framework for describing and realising pitch accent features. This framework was originally devised for use with the old HLCB intonation module. In that system, each element (H, L etc) had a default value for each variable on the numerical-RFC event level. A set of features were then defined which, when present for a particular accent, modified the default definition.

This feature system was very powerful, in that a feature could multiply, divided, add, subtract or reset any value in the element definition. Thus the order in which the features were applied could make a difference. This system did not allow for scalar quantities.

This system was amended for use with the tilt event level. Two major changes were that scalar "features" were allowed, and that the binary features were restricted in their power such that feature order wasn't important. There is of course nothing to say that the phonological system of English intonation is

feature based, but it should be clear that any discrete system can be reduced to a binary feature model if some of the feature combinations are disallowed. (e.g. Pierrehumbert's system can be reduced to a set of features such as "single/two tones", "high/low first tone", "high/low second tone" "first/second tone starred". The sixteen possible combinations quickly reduce to the permitted six, when it turns out many are the same and when one disallows the first and second tone being the same.)

As we have no independent way of specifying what the underlying description for an event should be, an automatic labelling system was developed. This system worked on normalised tilt variables. Normalisation simply involves mapping the each event in the data from Hertz and millisecond dimensions into z-scores. Z-scores are calculated by subtracting the mean of a variable and dividing by the standard deviation. The means and standard deviations of each variable were found from using all the labelled events from each speaker separately.

A set of features are then proposed and classified as being scalar or binary. The way each tilt variable is affected by each feature is defined. In all the experiments here, the binary features were simply defined as doing nothing to a tilt variable, or adding a score of +1 or -1 to a tilt variable. Scalar features take a z-score value in the underlying description, and modify the event according to a factor given in the feature definition. For example, a scalar "amplitude" feature was used whereby it modified the z-score of the amplitude of the event by the same value as the underlying event was specified for, and modified the z-score of the duration by half as much.

The automatic labeller labels each event in the utterance in turn by a analysis by synthesis method. First off all, the default values corresponding to the four mean tilt values for that speaker are used to defined the "test" event. Next the scalar values are assigned to this test event. This is done by measuring the values in the original event that are affected by each scalar feature. The test tilt event is then modified that the variables specified in the feature description have the same values as the variables of the original. Next, all combinations of features are applied to the test event, and each time the test event is compared to the original. The combination of features which gives the highest similarity score is chosen.

7.2 Which Features?

What features should we use? The number of possibilities is vast and so it will be impractical to test every one. It should be obvious that the more features one allows, the more accurate the system will be able to model F0 contours. And the more features one allows, the more verbose the feature-based event descriptions will become. In the case of scalar features, it should be clear that four are enough to accurately model the data without error (i.e. one scalar feature for each tilt variable). And it should also be clear that even a single scalar feature is very powerful.

The ideal situation would be to have a small number of features, with as few as possible being scalar, and for these features to be representative of the phonological structure of the language.

Many feature combinations are possible, and table 7.2 shows figures for 10 different feature combinations, ranging from all features being scalar to no features being present at all. The effect of the features on the tilt variables is shown in tables 7.2 and 7.2. The scores shown are between the feature level contours and the tilt level contours, the error between the tilt level and the RFC level is unchanged from table 7.2.

The general pattern is as one might expect with accuracy coinciding with descriptive power of the features. Again, many other feature combinations are possible, but these results are useful for showing the basic operation of the system.

As stated before, one criteria for choosing features is to look at the distributions of observations in the tilt space. Set C is the set of features currebtly used in the synthesizer.

It is clear that in figure 5, there is a clear tri-modal distribution in tilt. This pattern was also observed for the other speakers in the data. The values of -1 correspond to the fall-only events, such as the "low"

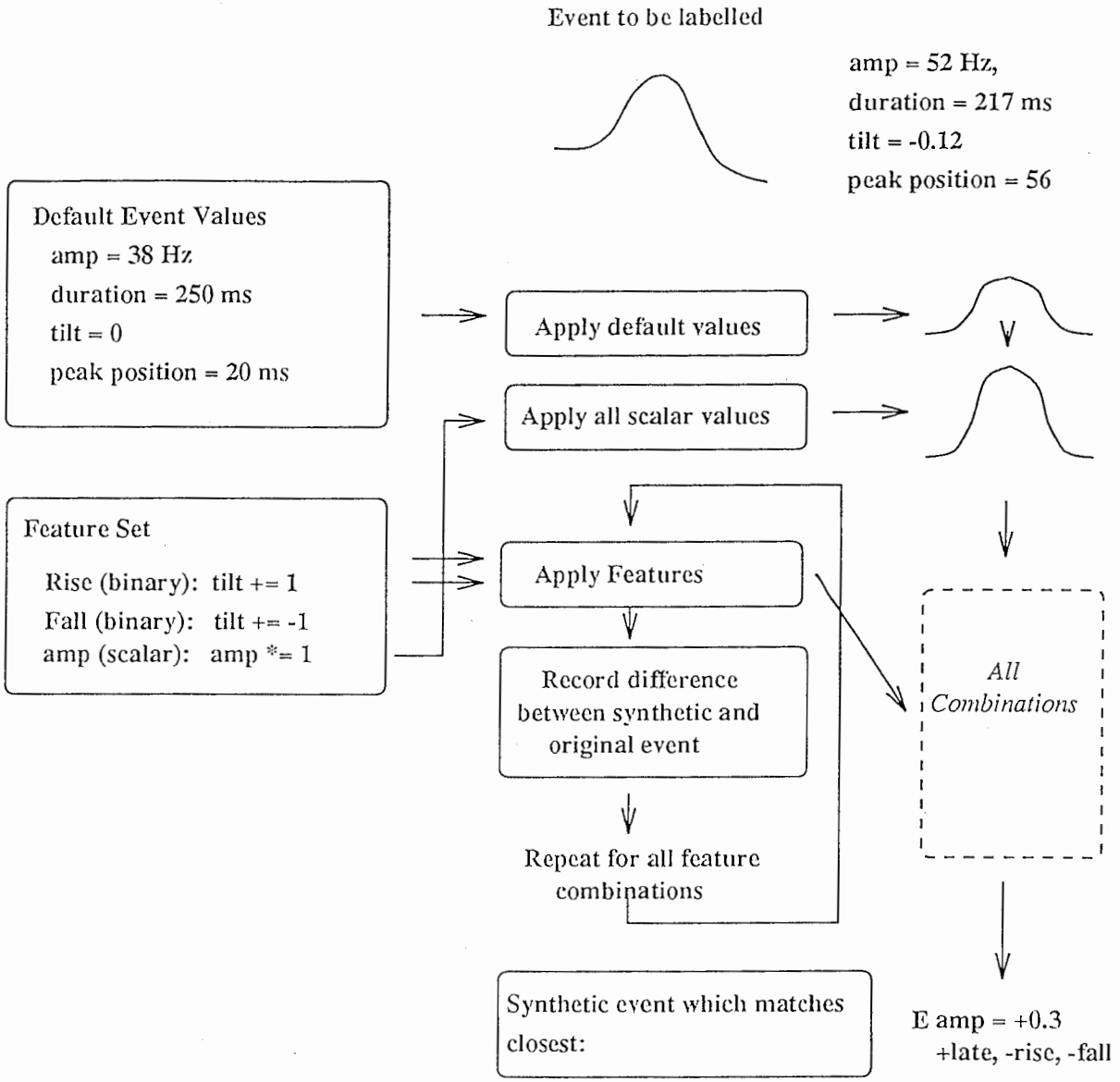


Figure 10: *The diagram on the left shows 3 parameters in tilt space. The diagram on the left shows 5 events for different values of tilt.*

Feature name	Affecting tilt variable	with value
rise	tilt	+ 1
fall	tilt	- 1
late	tilt	+ 1
early	tilt	- 1
big	tilt	+ 1
small	tilt	- 1
amp	amp	scalar
dur	dur	scalar
peak pos	peak pos	scalar
tilt	tilt	scalar

Table 2: *Event Features and their values*

Feature name	Affecting tilt variable	with value
rise	amp	+ 1
fall	amp	- 1
amp	amp	scalar

Table 3: *Connection Features and their values*

Experiment	Event binary features	Event scalar features	connection	error
A	-	amp, dur, tilt, peak pos	scalar	3.3
B	rise, fall	amp, dur, peak pos	rise, fall	5.3
C	rise, fall, early, late	amp	rise, fall	7.6
D	rise, fall, big, small, early, late	-	scalar	8.8
E	rise, fall, big, small, early, late	-	rise, fall	9.7
F	rise, fall, big, small	-	scalar	13.6
G	rise, fall, big, small	-	rise, fall	13.7
H	big, small	-	scalar	17.0
I	big, small	-	rise, fall	21.0
J	-	-	-	26.2

Table 4: *Accuracy of synthesis for different feature sets for one speaker (MAEM).*

Speaker	feature → RFC	feature → tilt	tilt → RFC
FALZ	16.0	13.4	5.25
FJMT	18.3	15.5	4.8
MAEM	11.1	8.5	4.8
MMAG	13.1	8.3	7.1

Table 5: *Accuracy of synthesis for four American speakers. The second column gives the average absolute distance between the synthesized contour and the contour generated from the hand-labelled numerical RFC level. The third column gives the distance between the contour generated from the feature level and the contour generated from the tilt level. The fourth column gives the error between the tilt-generated contour and the RFC-generated contour.*

pitch accents and downstepped accents. The values clustered in the centre represent the "normal" rise-fall pitch accents, and the values at +1 correspond to the rise-only events, such as the resets and the boundary rises. From this graph it seems justified to have a three way split in the feature description. The features *rise* and *fall* were proposed, with *rise* setting the the tilt value to +1, and fall setting it to -1. Thus +rise, +fall is the same as -rise,-fall, giving a total of three classes.

Amplitude is the strongest correlate of perceived prominence. As it is common to think of this as being a scalar quantity, and as the distribution does not show any n-modal distributions, amplitude was controlled in a scalar way.

The two remaining tilt variables, duration and peak position do not show any separation in their distribution either. However, it would be giving too much power to the feature description to allow these to be modelled in a scalar fashion also. Therefore two features were used for peak position in much the same way as the values were used for tilt. *Early* sets the zscore to -1, and *late* sets it to +1. *Early* is often marked in downstepped accents, and *late* is often found during rise-only events.

The correlation between duration and amplitude is higher than between duration and either tilt or peak position. Although a fixed-gradient system can not be truly advocated, for purposes of experiment, duration was matched to the amplitude scalar feature, such that an increase in amplitude produced a corresponding increase in duration.

Connections also show variation, and the features rise and fall were used to allow another 3 way division of the space.

Using these features, the automatic labeller labelled all the utterances for the 4 test speakers.

7.3 Discussion of Feature System

The feature labelling and synthesis system can analyse and synthesize contours with acceptable accuracy. The most used feature combination produced accuracies of between 11.1 and 18.3 Hz average distance between contours synthesized from the feature level and contours synthesized from the RFC level. Informal listening tests show that the feature-synthesized contours do sound very similar (but not identical) to the RFC-synthesized contours. Although improvement is still obviously possible, this amount of accuracy is acceptable for the intonation component of a speech synthesis system.

Although achieving high synthesis accuracy, the feature-based system as it now stands suffers badly from the arbitrary nature of the features themselves. The basic problem is that a wide variety of different feature sets could have the same accuracy. Although this is not a problem for generating F0 contours, it implies that the choice of feature system is somewhat arbitrary and that a higher level module aiming to generate the feature descriptions may have difficulty. In other words we have no reason to believe that the feature system is linguistically meaningful or relevant.

This is not really a surprise since a purely bottom up analysis forces this type of result. However,

the construction of the feature based system was a useful experiment as it showed that a small number of features are powerful enough to synthesize intonation.

8 The Synthesis Algorithm

The synthesis algorithm itself is shown in figure 11. The input to the system is a list of phrases, where each phrases contains a list of syllables. Each syllable has a list of phonemes, and each phoneme has an identifier and a duration. Attached to each syllable may be one or more events. Between each event is a connection. The events are labelled with the feature descriptions as defined in the feature set currently in use. The feature description for utterance C12.02 from speaker MMAG is shown below.

```
(Utterance  (Syllable (space tilt)(format feature)(dimen z)) (
(:C ((Start 0.750000))
   ((hh  60) (eh  65)                          ((C ) (E fall (amp -0.19) ) ))
   ((l   33) (ow 207)                          ((C ) ))
)
(:C ((Start -0.100000))
   ((dh  27) (ih  56)                          ((C ) (E (amp -0.10) ) ))
   ((s   75) (ih  56) (z   44)                 ((C fall) ))
   ((dh  42) (ax  36)                          ())
   ((k   95) (aa 129) (n   44)                 ((E fall (amp -0.14) ) ))
   ((f   77) (r   36) (en  57)                 ((C ) ))
   ((s   77) (ao 156)                          ())
   ((f   83) (eh 105) (s  203)                 ())
)
))
```

All scalar quantities are in z-scores. The following steps are performed.

1. Each event's features are looked up in the feature definitions table. Scalar features are applied first, and then binary features.
2. The events are mapped out of z space by using the speaker characteristics table. Each event's parameters are multiplied by the standard deviation and the mean is added.
3. Events are mapped from tilt space into RFC space by using the equations give in section 6.
4. The rises, falls and connections are mapped into F0 contours by the standard RFC equations.

9 Discussion

9.1 Summary of Main Findings

There were three main developments in the work reported here. Firstly it was shown how intonation could be described as a chain of phrases comprised of alternating sequences of events and connections. It was shown that a single formal specification could account for all intonational events. Secondly, the tilt system was introduced which transformed events in the RFC space to the tilt space. In doing so, the

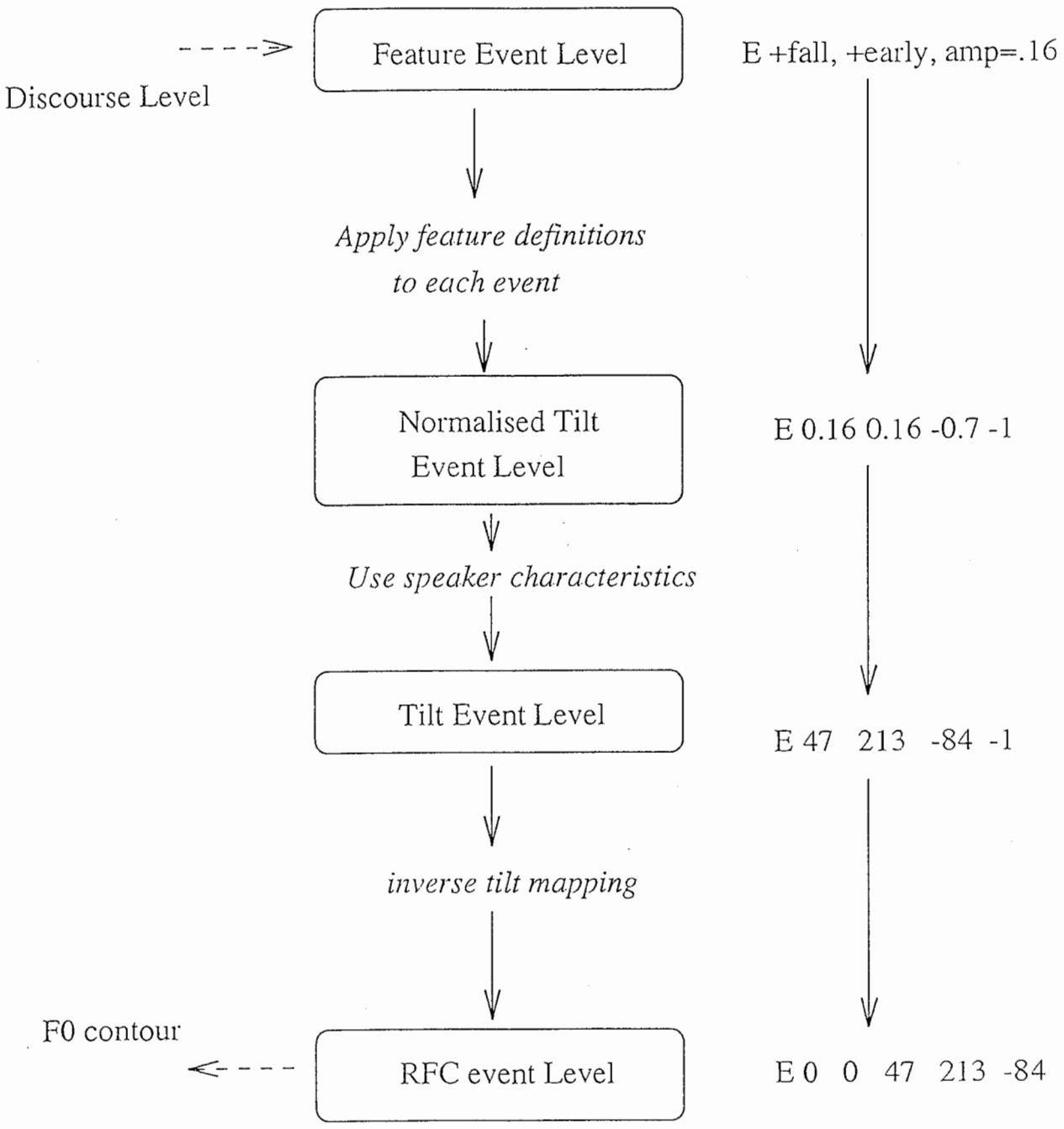


Figure 11: *Flow chart of synthesis algorithm.*

Variable	Mean	S. D.
amplitude	46	32
duration	316	161
tilt	0.0	0.78
peak position	-74	261
phrase start	222.0	25

Table 6: *Speaker characteristics for FALZ.*

number of variables needed to describe an event was reduced from 5 to 4, and importantly these new variables were more independent and linguistically meaningful. Finally, a feature system was proposed that could analyses events in the tilt space, classify them according to the feature specification, and synthesize them again. A wide variety of feature systems could be proposed and changed easily, thus allowing experimentation and testing of different ideas.

The event and tilt ideas proved to be very powerful and provide a solid foundation for future research. The feature-based system was less useful, as it suffered from the arbitrary specification of the features. However many useful insights were gained from these experiments.

9.2 Speaker Characteristics

Another important consideration in the work was that it proved easy to capture the intonational characteristics of each speaker. This was done by measuring the mean and variance of each of the four tilt variables and the pitch range. Thus 10 figures represent all the speaker dependent characteristics needed to map from the feature level to the F0 level. A set of typical values for a speaker is given in table 9.2.

These figures are easy to collect and thus give a very useful measure of a speaker's characteristics. More data from each speaker would help ensure that these figures are truly representative for the speaker.

9.3 Downstep and Declination

A useful feature of studying intonation by means of a formal model is that it allows inspection of many aspects of the intonational process. One of the more important studies conducted here concerns the issues of downstep and declination.

Downdrift is a name given to the observation that F0 contours generally end at lower F0 values than they start. Many researchers have tried to account for this phenomena. The two most important views are the declination account and the downstep account. The declination hypothesis says that downdrift is the inevitable consequence of speaking and is a low level effect accounted for by speaker's running out of breath and changes in vocal fold tension over time. As such it is a low level phonetic effect and not readily controllable by the speaker. Many have proposed "super-impositional" models whereby the linguistic intonation pattern is superimposed on "sloping graph paper" which accounts for the declination effect. Fujisaki (1982), Halliday (1967) and the Dutch school (t'Hart and Cohen, 1973) are proponents of this view.

The other main hypothesis is that declination as described above exists, but the main source of downtrend is due to the phenomena of *downstep* (Pierrehumbert, 1980), (Pierrehumbert and Beckman, 1991). The claim here is that downstep is a phonological effect which causes accents to be realised at lower levels than they otherwise would be. As it is a phonological effect, the speaker is in effect able to control the downdrift more easily than declination. A central and very controversial aspect of the original claim is that downstep is triggered by particular sequences of tones, and is thus an automatic consequence of tonal realisation. For practical purposes it is not particular relevant whether downstep is triggered or

Speaker	Events	Connections
FALZ	-28.1	-0.6
FJMT	-31.4	-0.76
MAEM	-15.1	0.0
MMAG	-25.3	-1.1

Table 7: *Contribution in Hz to downdrift from pitch accent events and from connections.*

is directly controllable, the important point is that the speaker has control. Pierrehumbert and colleagues still maintain however that there is a slight declination (i.e. phonetic) effect still in operation.

The event based model is ideal for the purposes of investigating these claims. First of all there is a clear separation between phonological entities (the events) and the remainder of the contour which is described by connections. Thus it is possible to assess the contribution to downdrift made by the events and connections separately and therefore discover how much of downtrend is attributable to downstep and how much to declination. Table 9.3 shows average difference between the start and end points of all the events and all the connection elements for each speaker.

The results are very clear, showing that downstep has a much stronger influence on downdrift than declination, which is practically negligible. Figure 12 shows the distribution of 100 connections elements from speaker FALZ. This graph clearly shows that not only is the mean downdrift effect nearly zero, but in fact the majority (52) of connection elements actually have zero downdrift. Figure 12 also shows however that some connection elements definitely do have steep negative gradients, but there are just as many with steep positive gradients. Similar distributions are seen for all speakers. Events are therefore by far the main contributors to downdrift, and it seems that declination as previously proposed has a very minor contribution.

There are two distinct types of pitch accent. downdrift observed in the data. For speaker MAEM, 74 accents out of a total of 141 have a tilt of -1, i.e. no rise at all. The average fall across these accents is 30 Hz. For the remaining pitch accents, the average tilt is -0.13, which corresponds to an average fall across the accent of 15 Hz. The first class typically have much earlier peak positions (average -86 ms) and correspond to downstepped and "Low" accents. The second class are the typical H or peak type of accent. It in interesting to observe that the first type of accent is much more prevelant in the American data than in any of the British data that was looked at in previous studies. This may be a key difference between these two accents of English.

9.4 Further Work

We are still some way from having a powerful phonological description, but a number of observations made in this study point the way forward.

Levels

From listening to the synthesized contours and comparing them with the originals, it was possible to develop a feel for what errors where perceptually most noticeable. It should be no surprise to say that the general shape of the F0 contour is important and an exclusively context free event system is not a natural way of describing intonation. In particular, it appears the ear is much more sensitive to differences in level, than say differences in timing.

The final F0 values are very important. This may be simply due to these values being more recent and therefore more memorable, but other studies have shown these values to be particularly important. Liberman and Pierrehumbert's study (1984) showed a large amount of invariance in the phrase final F0

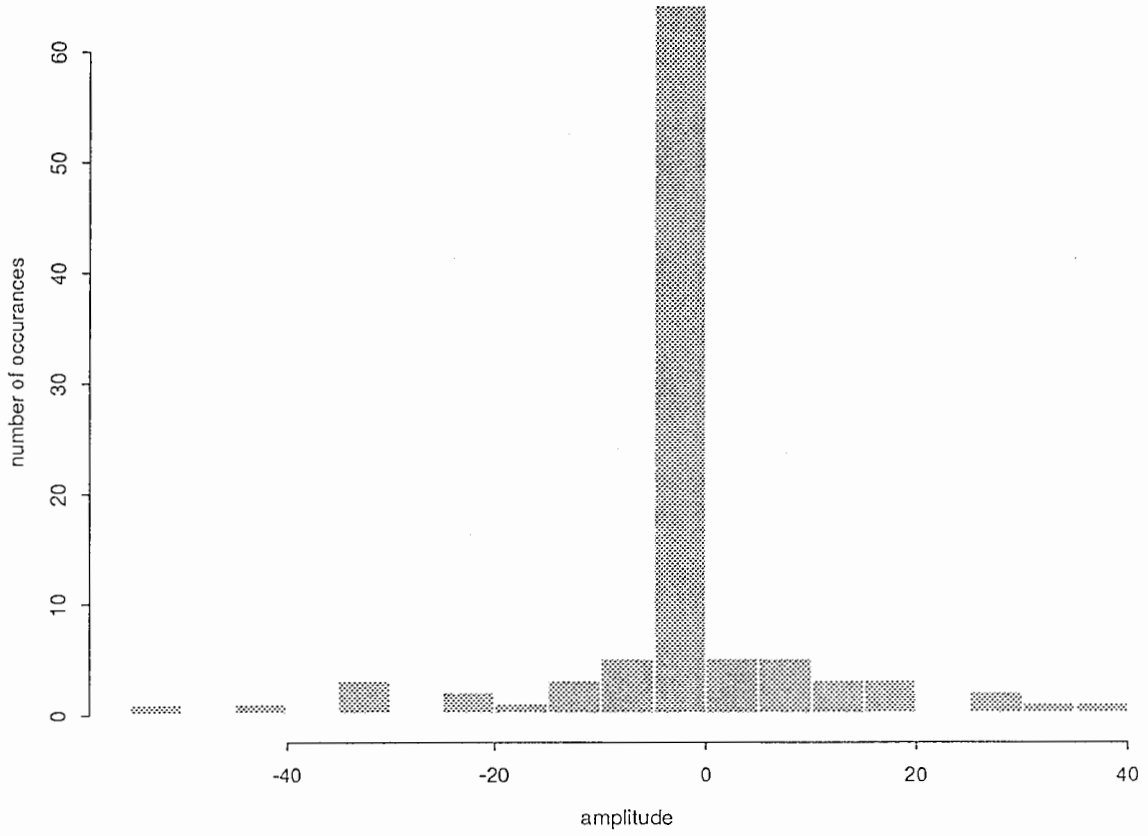


Figure 12: *Distribution of Connection amplitudes.*

values in their data such that they proposed a phenomena of phrase-final lowering. We do not have enough data of the right type to repeat this experiment, but it seems that phrase-final lowering effect is very important and must be modelled correctly to avoid unnaturalness.

These observations would lend support to a "levels" based analysis rather than a "dynamic" analysis of English intonation, which is of course *the* classic topic of debate in intonation. I think a reasonable explanation is that at a low level, intonation is a function of pitch movements, but that the amplitudes of these pitch movements are determined on a more tonal basis. Levels themselves are not the whole story: if one synthesizes a contour which interpolates between peaks and has no intervening dips, the quality is very poor.

Features

The feature system gave interesting results and shows that it was possible to accurately model events with a small number of features. It was also shown that an automatic feature labeller is a practical way of analysing events. The technique's failure stems from the fact that it is really just a knowledge based vector quantization routine.

As said previously, two approaches were advocated for the problem of mapping between the tilt level and a higher level. It is clear that the first approach need to be taken and more work should be devoted to the phonetic realisation module.

Testing

I have slight reservations about the testing procedures used here. It is not clear that directly comparing F0 contours is always useful. When contours are very similar (i.e. difference less than 10Hz), the measure is very fair, as it is basically stating the contours are identical for practical purposes [3]. However, when the difference is much greater, for example the error of 26 Hz for experiment J in table 7.2, the measure becomes less useful. This is basically because not all contours with an error of 26 Hz sound equally bad. The direct comparison measure should be retained, but only used as a real comparison between methods when the values are low.

[3] The human difference limen for F0 being about 5Hz.

Appendix A: Code

The Tilt synthesis algorithm runs in real time and is fully implemented in the Chatr speech synthesis system. All the analysis and all the synthesis routines described in this paper are user functions in Chatr. There are no separate C programs, awk code or hacky shell scripts.

The analysis programs were written within chatr as it provides support for the kind of complex data structures needed to implement this type of system. Furthermore, Chatr is a easy environment to work in and algorithm development is orders of magnitude quicker when using Chatr.

Within Chatr, the on-line documentation provides helpful advice to the user. Here we will give a few more details.

Whether performing analysis or synthesis, the basic operation is to transform an utterance from one level to another. To use one of these functions, the utterance must have at least been loaded and passed through the `Input` function.

Feature_to_Num Converts feature description to z-score numerical description.

Z_to_Linear Converts Z score event to numerical event in tilt space. The mean and s.d. or each tilt variable must have been previously using the `Stats Intonation` function.

Tilt_to_RFC Converts event in tilt space to event in RFC space.

Rfc_Module Creates F0 contour from RFC description.

Feature_Int Converts from Features to RFC in one module.

Rfc_to_Tilt Analysis routine which converts event in RFC space to event in tilt space.

Linear_to_Z Normalises event in Tilt space according to current intonation stats, defined by `Stats Intonation`.

Num_to_Feature Converts z score event to feature event using feature table (defined again using `Stats Intonation`.

Int_Stats Returns list of event and connection descriptions for an utterance.

Syllabify Creates a syllable stream from the phoneme stream using the sonority and maximal onset principles. This is use to make syllable input files from segment or RFC input files.

Phrasify Creates a prosodic phrase stream from the phoneme stream using pauses in the phoneme stream to indicate phrase boundaries. This is used to create syllable input files from segment or RFC input files.

.1 Chatr language functions

A number of lisp chatr functions exist to help in the processing of syllable utterances. These functions along with documentation are currently in the file tilt_lib.ch.

Appendix B: Adding a New Speaker

The most difficult part about adding a new speaker is labelling the data. Once the data is in the form that Chatr requires, everything else is simple.

Chatr requires a syllable utterance type description for each utterance. This is comprised of a list of phrases, each with a start F0. Within each phrase is a list of syllables and each may have one or more events marked. An example is given below:

```
(Utterance  (Syllable (space rfc)(format feature)(dimen num)) (
(:C ()
   ((hh  60) (eh  65)                          ((E)))
   ((l   33) (ow 207)                          ())
)
(:C ()
   ((dh  27) (ih  56)                          ((E)))
   ((s   75) (ih  56) (z   44)                 ())
   ((dh  42) (ax  36)                          ())
   ((k   95) (aa 129) (n   44)                 ((E)))
   ((f   77) (r   36) (en  57)                 ())
   ((s   77) (ao 156)                          ())
   ((f   83) (eh 105) (s  203)                 ())
)
))
```

In this type of description, only the presence of an event need be marked.
In addition, an RFC input description is required. An example is given below.

```
(Utterance RFC(
(sil 303 ( ( sil 0 166 ) ))
(hh 60 ())
(eh 65 ( ( fall 21 166 )))
(l 33 ())
(ow 207 ( ( conn 67 125 ) ( sil 197 120 )))
(sil 155 ())
(dh 27 ( ( rise 0 149 )))
(ih 56 ())
(s 75 ( ( fall 60 173 )))
(ih 56 ())
(z 44 ())
(dh 42 ( ( conn 4 151 )))
(ax 36 ())
(k 95 ())
(aa 129 ())
(n 44 ( ( fall 5 142 )))
(f 77 ())
(r 36 ())
(en 57 ())
(s 77 ( ( conn 74 95 )))
(ao 156 ())
(f 83 ())
(eh 105 ())
(s 203 ( ( sil 91 91 )))
```

```
(sil 524 ())
))
```

The Chatr user function `train_input` takes these two utterance descriptions and produces a syllable description inthe RFC event space:

```
(Utterance  (Syllable (space rfc)(format num)(dimen linear)) (
(:C ((Start 166))
   ((hh  60) (eh  65)
 ((C   0.00) (E 0.00 0.00 -41.00 144.00 21.00)))
   ((l   33) (ow 207)                          ((C   0.00) ))
)
(:C ((Start 149))
   ((dh  27) (ih  56)
 ((C   0.00) (E 24.00 143.00 -22.00 119.00 116.00)))
   ((s   75) (ih  56) (z   44)                 ((C  -9.00) ))
   ((dh  42) (ax  36)                          ())
   ((k   95) (aa 129) (n   44)
 ((E 0.00 0.00 -47.00 283.00 134.00)))
   ((f   77) (r   36) (en  57)                 ((C  -4.00) ))
   ((s   77) (ao 156)                          ())
   ((f   83) (eh 105) (s  203)                 ())
)
))
```

Next, the function `Rfc_to_Tilt` is called which transforms this into tilt space. With a sufficient number of utterances in tilt space, statistics can be collected on each of the 4 tilt parameters and the phrase start F0 parameter. The mean and s.d. needs to be calculated which can be done using S or any other utility. The tilt descriptions can be derived from the utterance file, or by using the `Int_Stats` function which returns a list of all the events, all the connections or both for an utterance. By calling mapc, one can get all the stats for a database.

Once the stats have been collected, one can construct a speaker table by filling in the mean and standard deviations in the appropriate places. A typical speaker file is given below:

```
(Stats Intonation (
(Element E (def tilt E)(
 (amp       = 47  Hz)
 (dur       = 291 ms)
 (tilt      = 0.0 rel)
 (peak_pos  = 59  ms)
))
(Element E (var tilt E)(
 (amp       = 31  Hz)
 (dur       = 141 ms)
 (tilt      = 0.75 rel)
 (peak_pos  = 136 ms)
))
(Element C (def any C)(
(amp        = 0.0 Hz)
```

```
))
(Element C (var any C)(
 (amp       = 10 Hz)
))
(Element P (def any P)(
(amp        = 151.0 Hz)
))
(Element P (var any P)(
 (amp       = 20 Hz)
))
))
```

Appendix C: Defining a New Feature Set

A little care needs to be taken here as completely stupid feature sets might confuse the system.

An example feature set is given below.

```
(Stats Intonation (
(Feature rise (binary tilt C)(
 (amp        +=  10 rel)
 ))
(Feature fall (binary tilt C)(
 (amp        -=  10 rel)
 ))
(Feature amp (scalar tilt E)(
 (amp        +=  1 rel)
 (dur        +=  1 rel)
 ))
(Feature early (binary tilt E)(
 (peak_pos -= 1.1 rel)
 ))
(Feature late (binary tilt E)(
 (peak_pos += 1.1 rel)
 ))
(Feature rise (binary tilt E)(
 (tilt       += +1 rel)
 ))
(Feature fall (binary tilt E)(
 (tilt       += -1 rel)
 ))
)
)
```

Feature headers are defined in the form

```
<name>   ( <type > <space> <element>)
```

The name need only be unique to the space and element, so that connection features and event features can have the same name without confusion. Type refers to scalar or binary, space refers to rfc or tilt, though I think only tilt is fully implemented. Element refers to whether the feature should operate on an Event or a Connection.

Feature bodies are defined in the form

```
(variable operator value dimension)
```

Variable specifies which tilt variables are to be affected, operator should always be += or -=, value is in s.d. so very large values are inadvisable. Dimension is not used.

xvm

References

Allen, J., Hunnicut, S., and Klatt, D. (1987). *From Text to Speech: the MITalk System.* Cambridge University Press.

Chomsky, N. and Halle, M. (1968). *The Sound Pattern of English.* Harper and Row.

Crystal, D. (1969). *Prosodic Systems and Intonation in English.* Cambridge Studies in Linguistics. Cambridge Universtity Press.

Fujisaki, H. and Kawai, H. (1982). Modeling the dynamic characteristics of voice fundamental frequency with applications to analysis and synthesis of intonation. In *Working Group on Intonation, 13th International Congress of Linguists, Tokyo.*

Halliday, M. A. K. (1967). *Intonation and Grammar in British English.* Mouton.

Ladd, D. R. (1983). Phonological features of intonation peaks. *Language*, 59:721–759.

Liberman, M. and Pierrehumbert, J. (1984). Intonational invariance under changes in pitch range and length. In Aronoff, M. and Oehrle, R. T., editors, *Language Sound Structure.* MIT Press.

Liberman, M. Y. (1975). *The Intonational System of English.* PhD thesis, MIT. Published by Indiana University Linguistics Club.

Monaghan, A. (1991). *Intonation in a Text to Speech Conversion System.* PhD thesis, CSTR.

O'Connor, J. D. and Arnold, G. F. (1973). *Intonation of Colloquial English.* Longman, 2 edition.

Palmer, H. (1922). *English Intonation with Systematic Exercises.* Cambridge Universtity Press.

Pierrehumbert, J. B. (1980). *The Phonology and Phonetics of English Intonation.* PhD thesis, MIT. Published by Indiana University Linguistics Club.

Pierrehumbert, J. B. and Beckman, M. E. (1991). *Japanese Tone Structure.* MIT press.

Silverman, K. (1993). Unpublishedtalk about prosody in speech synthesis. In *ESCA Workshop on Prosody, Lund, Sweden.*

Taylor, P. A. (1992). *A Phonetic Model of English Intonation.* PhD thesis, University of Edinburgh.

Taylor, P. A. (1993). Synthesizing intonation using the RFC model. In *Proc. ESCA Workshop on Prosody, Lund, Sweden.*

t'Hart, J. and Cohen, A. (1973). Intonation by rule: a perceptual quest. *Journal of Phonetics*, 1:309–327.

The Rise/Fall/Connection Model of Intonation

Paul Taylor
ATR Interpreting Telecommunications Labs

ATR Interpreting Telecommunication Laboratories,
2-2 Hikaridai,
Seika-cho,
Soraku-gun,
Kyoto,
JAPAN

email
paul@itl.atr.co.jp

Abstract

This paper describes a new model of intonation for English. The paper proposes that intonation can be described using a sequence of rise, fall and connection elements. Pitch accents and boundary rises are described using rise and fall elements, and connection elements are used to describe everything else. Equations can be used to synthesize fundamental frequency (F_0) contours from these elements. An automatic labelling system is described which can derive a rise/fall/connection description from any utterance without using prior knowledge or top-down processing. Synthesis and analysis experiments are described using utterances from six speakers of various English accents. An analysis/resynthesis experiment is described which shows that the contours produced by the model are similar to within 3.6 to 7.3 Hz of the originals. An assessment of the automatic labeller shows 72% to 92% agreement between automatic and hand labels. The paper concludes with a comparison between this model and others, and a discussion of the practical applications of the model.

Résumé

Nous présentons dans cet article un nouveau modèle d'intonation pour l'anglais, selon lequel celle-ci est décrite en termes "descentes"(rise), "montées" (fall) et "lignes de connection" (connection). Les accents et les montées de continuation sont représentés par les éléments de "montée" et de "descente" ; les lignes de connection sont utilisées partout ailleurs. Un système d'équations permet de reconstruire le contour de fréquence fondamentale à partir de ces éléments. Nous décrirons ensuite un système d'étiquetage automatique qui associe à toute phrase une représentation descentes/montées/connections sans connaissance a priori, ni analyse descendante. Une série d'expériences portant sur un corpus de phrases prononcées par 6 locuteurs de langue anglaise et d'accents variés valide ce modèle : l'erreur de synthèse est comprise entre 3.6 et 7.3 Hz. Par ailleurs, une comparaison des étiquetages automatiques et manuels montre une correspondance de 72% a 92%. Nous conclurons par une comparaison du modèle proposé et des modèles existants et en discuterons les applications pratiques.

Zusammenfassung

Wir stellen ein neues Model für englische Intonation vor. Wir gehen davon aus daß Intonation sich durch eine Reihe von Steig-, Fall- und Verbindungselementen beschrieben läßt. Pitchakzente und Grenzsteigerungen werden durch Setig- und Fallelemente beschrieben und Verbindungselemente werden benutzt um alle andere Phänomene zu beschreiben. Die Fundamentalfrequenzkonturen können mittels Gleichungen von diesen Elementen berechnet werden. Ein automatisches Markiersystem wird beschrieben mit dem die Steig/Fall/Verbindungselementkette automatisch one vorheriges Wissen oder top-down Berechnung erstellt werden kann. Synthese- und Analyseexperimente sind beschrieben für Äußerungen sechs verschiedener Sprecher unterschiedlicher Akzente. Ein Analyses-Resyntheseexperiment ist beschrieben was zeigt daß die Konturen um zwischen 3.6 bis 7.3 Hz vom Original abweichen. Ein Bewertung des automatischen Markiersystems zeigt daß etwa 72% bis 92% Übereinstimmung zwischen den automatischen und von Hand geschriebenen Markierungen besteht. Ein Vergleich mit anderen Modellen und eine Beschreibung wie das Modell benutzt werden kann beenden dieses Paper.

1 Introduction

This paper describes a new model of intonation which accurately synthesizes fundamental frequency (F_0) contours from a linguistically relevant description. An automatic labelling system is described which can derive this linguistically relevant description from any F_0 contour.

Fundamental Frequency (F_0) synthesis algorithms are often based on the principle of describing a set of prototypical F_0 patterns which represent different intonational tunes of the language in question (Isard and Pearson, 1988), (Anderson et al., 1984), (Willems, 1983), (Vonwiller et al., 1990), (Ladd, 1987). Other researchers analyse a database of F_0 contours and use statistical techniques to learn the realisation of intonational tunes (Sagisaka and Kaiki, 1992), (Traber, 1990). There are fewer systems devoted to the inverse problem of deriving a symbolic or

parametric description from an F_0 contour, although recently more systems have been reported (Jensen et al., 1993), (ten Bosch, 1993).

A different approach is to use an *intonation model*, which is a general device for relating F_0 contours to a higher level intonational description. Fujisaki (1988), t'Hart & Cohen (1973) and Hirst (1992) report work in developing models of this kind. Here we propose a new intonation model for English and show how this model can be used for the analysis and synthesis of F_0 contours.

Section 2 describes the model itself, and section 3 describes an automatic labelling system which can derive the model's parameters from F_0 contours. Section 4 describes the data we used to test the new model with, and sections 5 and 6 discuss the synthesis and analysis capabilities of the model. Section 7 discusses the need for flexibility when modelling F_0 and compares the new model with previous work.

2 The RFC Model

The model presented here, termed the *Rise/Fall/Connection* Model (RFC for short), was not developed with a particular phonological theory in mind. However, as it is the end goal of this research to eventually provide a complete mapping between acoustics and phonology, it is necessary to show that the RFC descriptions produced by the model do in fact make a substantial contribution to the solution of the overall problem. One can see from a survey of the literature that phonological theories of intonation largely agree on what phenomena need to be described, even if there are differences in the details of classification. Most contemporary theories agree that phonological descriptions of English intonation are primarily concerned with describing behavior of *pitch accents* on stressed syllables and *boundary tunes* at prosodic phrases edges. These theories also state that there are distinct classes of pitch accent and boundary tunes which have distinctive F_0 patterns. These two basic types of phenomena operate with respect to different prosodic units: pitch accents are associated with syllables and boundary tunes are associated with the beginnings and ends of prosodic phrases. This view is more or less consistent with the work proposed by Pierrehumbert (1980), O'Connor and Arnold (1973), Halliday (1967), Ladd (1983) and others[1]. Grice (1992) reviews and compares these and other phonological theories.

2.1 Modelling F_0 Contours with Equations

From examination of the large number of pitch accents in our data (described in section 4), it was clear that there were two basic types of pitch accent. The first type were "peak" or "high" accents, in which the accented syllable was associated with a peak in the F_0 contour. The other type were "trough" or "low" accents, where the converse effect was obvious - i.e. the accented syllable was associated with a trough in the F_0 contour[2]. Peak accents roughly correspond with the H* class of Pierrehumbert (1980), the H class of Ladd (1983) and the "fall" class of O'Connor and Arnold (1973). Trough accents are roughly equivalent to the L* class of Pierrehumbert, the L class of Ladd and the "low" class of O'Connor and Arnold.

Peak Accents

Using a empirical trial and error study, an equation was devised that can accurately synthesize any pitch accent with a few parameters.

Peak accents are modelled by describing the rise and fall parts of the accent separately. Equation 1 (termed the *monomial* equation) is used to model both rise and fall. The form of the equation shown here describes the F_0 of the fall part of a peak accent: when this curve is reflected in the y axis it describes the rise part. In our data there was a large variation in the amplitudes and durations of the rises and falls (both absolutely and relative to one another) and there was also a wide variation in the gradients of these rises and falls. To model this variation, two scaling

[1] Pierrehumbert also includes the "phrase accent" as a fundamental unit of intonation. This is not problematic to our baseline notion in that the phrase accent can be included in the general classification of boundary tune, as phrase accents are realised only after the nuclear accent. O'Connor and Arnold make a four way classification of fundamental units. Following from Pierrehumbert and others, we include "head" and "nucleus" to be of the same basic type (pitch accents), and also classify "tail" and "pre-head" as being describable by a mechanisms of boundary tune.

[2] There were a few cases which did not obviously fall into either of these two types as there was no noticeable F_0 pattern at all that could be linked to the perceived accented syllable. Such "level" pitch accents obviously deserve attention in any phonological framework, but as the work presented here is primarily concerned with characterising surface F_0 patterns rather than giving a phonology of pitch accents, we can ignore these cases for the time being.

factors are applied to the equation, allowing the equation to be stretched in the x and y dimensions. Equation 2 describes the formula with the scaling variables A and D, and figure 1 shows a plot of this function.

Although the rise and fall parts are given by the same equation, these parts still have to be modelled separately as there was no observable simple relation between the amplitude, duration or gradient of the rise and fall parts of an accent.

$$\begin{aligned} y &= 1 - 2.x^2 \quad 0 < x < 0.5 \\ y &= 2.(1-x)^2 \quad 0.5 < x < 1.0 \end{aligned} \tag{1}$$

$$\begin{aligned} f_0 &= A - 2.A.(t/D)^2 \quad 0 < t < D/2 \\ f_0 &= 2.A.(1-t/D)^2 \quad D/2 < t < D \end{aligned} \tag{2}$$

where A is element amplitude and D is element duration.

These rises and falls are termed *elements*, thus peak accents are modelled as a *rise element* followed by a *fall element*. The amplitude and duration scaling factors are called the element *parameters*. Often in a peak accent no rise is present, so the pitch accent is modelled using only a fall element. The phenomenon known as "flat hat" accents in the Dutch school (t'Hart and Cohen, 1973) could either be interpreted as two accents, one with a single rise element, the other with a single fall element; or as a single accent, where the rise and fall elements were separated by a straight section of F_0 contour.

Trough Accents

Trough accents are also modelled by the monomial function. The basic pattern for these accents is the reverse of the peak accent, i.e. a fall element followed by a rise element. Often a trough accent only needs a single rise or fall element.

Boundary Rises

Sharp rises are commonly found at both phrase beginnings and ends, and these are modelled using rise elements. Phrase initial rises are often termed *declination resets* (Ladd, 1988) as they serve to make sure the starting F_0 of the phrase is higher than the F_0 values of the previous phrase. Phrase final rises usually either give the impression of a continuation rise, indicating that more information is to follow, or of being part of a complex nucleus/tail relationship, where they often give the precept of a question.

Between Rise and Fall Elements

Not every part of an F_0 contour is comprised of pitch accents and boundary rises. Often there are parts where nothing of intonational interest is occurring. In these areas a straight line is used to model the F_0 contours. This element is termed the *connection element*. The connection element also has variable duration and amplitude parameters.

2.2 Well-Formedness Conditions

Thus in our model, F_0 contours are modelled with a linear sequence of rise, fall and connection elements, each of which can have any amplitude or duration. A set of well-formedness conditions are used to constrain this system:

- Pitch accents are modelled using at most a single rise and a single fall element.
- Boundary rises are modelled using a single rise element.
- Connection elements are used to model everything else in the F_0 contour, thus rise and fall elements may only be used for modelling pitch accents and boundary rises.
- Only one connection element is allowed between rise and fall elements.

An example RFC description as hand labelled from an F_0 contour in the test data is given in table 1, and the synthesized F_0 contour from this description is shown later in figure 3.

3 Automatic Labelling of F_0 contours using the RFC model

The system described below automatically labels F_0 contours in terms of the rise fall and connection elements. It works on unconstrained input: any F_0 contours can be labelled by this system without prior knowledge of the content of the utterance. Although there are trainable parameters in the system which give better performance when adapted for particular speakers, this system can label any type and any length of F_0 contour from any speaker of English.

3.1 F_0 Extraction

F_0 contours are usually extracted from speech waveforms using *pitch detection algorithms* (PDAs). The PDA used here was the super resolution pitch detection (SRPD) algorithm originally developed by Medan et al. (1991) and implemented by Bagshaw et al. (1993).

In normal usage the PDA produces F_0 contours which are influenced by segmental effects such that the immediate segmental environment affects the local shape of the F_0 contour. *Unvoiced segments* are the most noticeable effect: during such segments there is no fundamental frequency at all. *Obstruent perturbations* cause sudden spikes and glitches, often at the boundary between a voiced and unvoiced part of the F_0 contour. It is desirable to normalise for these affects because they are purely a result of the segmental environment and play no role in determining the underlying intonational tune of the utterance. To put it another way, two utterances with the same intonational tune can have apparently different F_0 contours solely due to these utterances having different segmental content. This problem was solved by adding a post-processing module to the PDA that converts the normal output into F_0 contours which are as free from segmental influence as possible, while maintaining exactly the same underlying intonational content.

In all the experiments described here, F_0 contours were specified at regular 5ms intervals. The first stage in the post-processing is to perform a 15 point median smoothing on the 5ms contour. This removes most of the obstruent perturbations and also the small scale effects of pitch perturbation or jitter that arise from the normal behaviour of the glottis. Next, the unvoiced regions are filled using straight line interpolation. Finally, a further 7 point smoothing is used to remove the occasional sharp edges at the boundary between the existing and interpolated parts of the F_0 contour. This processing is shown in figure 2.

This post-processing removes much of the segmental influence without unduly affecting the global shape of the F_0 contour. Heavier smoothing removes more of the segmental influence this can cause the global shape of the F_0 contour to change, thus potentially distorting the intonational information

3.2 Broad Class Labelling

The *broad class labelling module* takes the post-processed F_0 contours and locates rise, fall, and connection elements. This module distinguishes rises from falls by the principle that all rises must have positive gradient and all falls must have a negative gradient. The rise and fall elements are distinguished from connection elements on the basis that these elements have steeper gradients than those of connection elements.

The F_0 contour is re-sampled at 50ms intervals as it is at those sorts of distances and above that pitch accent rises and falls are realised. Next, the F_0 of each frame is compared with the F_0 of the previous frame, and if this exceeds a threshold, the frame is labelled a rise, and if it is below another (negative) threshold, the frame is labelled as a fall. All frames between these two thresholds are left unlabelled. These thresholds (termed the *rise gradient threshold* and *fall gradient threshold* respectively) are trained by the method described in section 6.2. At this stage, every frame is labelled either rise or fall, or is left unlabelled. Frames which have the same labels as their neighbours are grouped together, dividing the F_0 contour into approximately marked rise and fall elements.

The labelling produced by this method correctly identifies most of the rise and fall elements in the F_0 contours. As mentioned above, the F_0 post-processor does not remove all obstruent perturbations, and occasionally these are mislabelled as rise or fall elements. These spurious labels have characteristically short durations. Using this fact, a *deletion module* was developed, whereby elements below a certain minimum duration are deleted. This *deletion threshold* is set by the training method described in section 6.2. This module greatly reduces the number of spurious elements.

The labelling produced by this system has divided the F_0 contour into rise, fall and unlabeled elements, and by implication, the pitch accents and boundary rises have been located. However, the boundaries between these

elements are only accurate to the size of the frame, i.e. 50ms. It was desirable to go further and find the exact boundaries of the elements as these are crucial in distinguishing different types of pitch accent. Precise boundary adjustment is performed by the *optimal matching module*.

3.3 Optimal Matching

Using the labels produced by the above procedure, the original 5ms sampled F_0 contour is analysed to determine the precise boundaries of the rise and fall elements.

Around each approximate boundary, a *search region* is defined. This extends an absolute amount, typically 0.15 seconds outside the approximately marked boundary, and a percentage distance, typically 20%, inside the approximately marked boundary. Thus for a fall element, a region is defined starting 0.15 seconds before the start of the fall and extending 20% into the fall, and another region is defined starting 20% from the end of the fall and extending 0.15 seconds after the end of the fall. To determine the precise boundaries, every possible fall shape that can be defined as starting and ending in these areas is synthesized and compared to the F_0 contour. The shape showing the lowest euclidean distance between itself and the F_0 contour is chosen as the correct shape, and its start and end position determines the precise boundaries of the fall element. Likewise for rise elements.

In principle, the size and position of the search regions could have been trained, but it was found that in all cases a search region corresponding to 150ms outside the approximately marked element and 20% inside the element is large enough to insure that the precise element boundaries will be found. It was very noticeable that so long as some variation from the approximately labelled boundaries is allowed, the system always chooses the same start and end positions. This proves that the start and end positions are not arbitrarily marked, as the same positions are consistently chosen by the system independently of where the search regions are defined.

With the optimal matching process complete, all remaining unlabelled sections are labelled as connection elements. Thus the entire F_0 contour is now labelled in terms of the RFC system.

4 Data

Six sets of data from American, English and Irish male and female native speakers of English were used to test the model.

Data set A from a male speaker (Northern Ireland accent) comprised of 64 carefully spoken sentences that cover the major tune types described in O'Connor and Arnold (1973). Data set B was from a male speaker (southern English RP accent) and comprised of 45 utterances from email and Unix news articles. Data sets C to F were from the ATR-CMU conference registration database, which simulated conversations between receptionists and guests at conferences. This data was spoken by 2 male (C and E) and two female speakers with Standard American accents[3].

This data totalled 231 utterances, within which there were 1654 rise and fall elements and 533 intonational phrases, as hand labelled. The utterances varied in length from a single word to 40 words in length. There was a wide variation of intonational tune types due to the conversational nature of much of the speech.

5 Synthesis Assessment and Results

An analysis/resynthesis test was used to measure synthesis accuracy. For each utterance in the database, its F_0 contour was analysed (automatically and by the hand labelling method described in section 6.1) and an RFC description was produced. From this, an F_0 contour was synthesized and compared to the original.

Subjective tests are sometimes used to assess the intonation component of speech synthesis systems. In these tests, listeners are played re-synthesized pairs of utterances, one with the original F_0 contour and one with the synthesized F_0 contour, and if listeners cannot distinguish the two, then the synthesized F_0 contours are deemed to be perceptually equivalent to the originals.

It is not clear that this sort of subjective evaluation is the most suitable way of assessing the synthesis accuracy of a model such as ours. Here a different approach was taken and *objective* tests were used, whereby an algorithm assesses the differences between the original and synthesized F_0 contours. This was because:

[3] Data sets A and B are described fully in Taylor (1992), and the ATR-CMU data is described in Wood (1992).

- It is unclear how meaningful yes/no decisions are on the acceptability of synthesized F_0 contours. F_0 contours are specified in the continuous frequency domain, and as with any continuous variable, it is somewhat nonsensical to compare for direct equivalence. When comparing any real numbers, it makes sense to state how close they are, rather than give a yes/no decision of equivalence.
- As noted by Hirst (1992), "a model which seemed perfectly adequate for LPC diphone synthesis may appear less satisfactory when used with very high quality speech synthesis such as that provided by PSOLA". Thus different waveform synthesis techniques can influence the judgment of listeners. We would not want to run the risk of claiming that our model can produce acceptable F_0 contours to have a later more sophisticated waveform synthesis technique to prove otherwise. The assessment of F_0 synthesis should be independent from waveform synthesis techniques.
- Objective tests are very practical in that a large amount of data can be easily tested. Other researchers can easily compare the synthesis accuracy of their model with that of ours.

Thus the synthesis assessment method that was used was one of direct comparison between original and synthesized F_0 contours. Each point in the original F_0 contour is compared with the point at the equivalent time frame in the synthesized version, and the euclidean distance is measured. A score is produced by summing the values for an utterance and dividing by the total number of frames for that utterance. This score gives a measure of the average differences between two F_0 contours, or to put it another way, at any given point the expected difference between the F_0 contours is given by this score. Table 2 gives the scores for the 6 data sets.

These results clearly show that the synthesized F_0 contours are very similar to the originals. Many of the bad scores are due to PDA problems such as pitch doubling, which causes a large comparison error. Thus the scores would be better if a more accurate PDA was used.

To put the figures in table 2 in perspective, it is worth making some points on the accuracy of PDAs and the difference limens of F_0 changes in humans. Hess (1983) gives a review of both these issues and describes several experiments on how sensitive humans are to F_0 changes in speech. Figures of between 0.3-0.5% for pure synthetic vowels and 4-5% for natural speech are given. The 4% figure measured at 195Hz, giving an absolute value of 7.8Hz, and the 5% figure was measured at 150Hz, giving 7.5Hz. Thus our synthesis results are close to the difference limen for natural speech.

Rabiner et al. (1976) discuss the assessment and accuracy of PDAs, but perhaps the most useful and up to date study is that of Bagshaw et al. (1993). Their experiment examined the output of 6 PDAs, including the super resolution pitch detection algorithm used here, and compared them to F_0 contours directly measured by a laryngograph. The average error of these algorithms varied from 1.78 Hz to 3.25 Hz for male speech, and from 4.14 Hz to 6.39Hz for female speech, with the SRPD algorithm giving the best results[4]. The results for the 4 males speakers and female speaker E are close to a few Hertz of the accuracy of the SRPD algorithm, and the results for the female speaker F are within the margin of error.

6 Automatic Labeller Assessment and Results

6.1 Hand Labelling and Analysis Assessment

The main problem with analysis testing is not so much in the exact method of testing but rather the difficulty in determining the correct labelling for an F_0 contour. The only practical way of assessing the automatic labeller's performance is to compare the labels it produces with those of a human labeller. The problem with such a method is that it relies on the ability of the human labeller. Human labelling is always prone to some inconsistency and arbitrariness no matter how expert the labeller. However, it is clear that some human made decisions are consistently more reliable than others, for instance, it is often much easier to determine the *location* of a pitch accent than trying to decide what the *type* of the pitch accent is.

Bearing this in mind, the database of F_0 contours was hand labelled according to the following criteria. First, boundary rises and any syllables that were judged to be accented were marked. Next, rises and falls were fitted to the pitch accents and boundary rises. Although the different types of pitch accent would naturally be expressed by their different rise and fall labellings, no explicit labelling of pitch accent class was performed.

[4] Bagshaw et al. go on to explain improvements to the SRPD which give better results than the original described in Medan et al. (1991). This improved version was not available for the experiments described here.

Determining the element parameter values was quite straightforward as the correct values were chosen to be the ones which made the synthesize F_0 contour most similar to the original. This was a process of trial and error, but nearly always a satisfactory match was found. Also, for a particular accent it was not the case that a wide variety of parameter values fitted equally well, usually there was a fit that was unarbitrarily the best.

The analysis assessment method calculates the percentage of rises and falls correctly identified. Thus the system simply counts the number of insertion and deletion errors, and divides this by the total number of tokens, resulting in a percentage recognition score.

6.2 Training the Automatic Labeller

A number of adjustable thresholds operate in the analysis system. The most important are the rise and fall *gradient* thresholds in the broad class labeller. These are used to determine whether a 50ms frame is to be labelled as a rise, fall or left unlabelled. In addition there are two *deletion* thresholds which specify the minimum allowable size for an element. Any element identified by the broad class labeller which is below this duration is deleted. Although the rise and fall thresholds operate independently of each other, the gradient thresholds interact with the deletion thresholds.

The training procedure operates by systematically adjusting the thresholds until the optimal set is obtained. Taking the case of the fall thresholds, a 2 dimensional table is built with one dimension representing the gradient threshold and the other representing the deletion threshold. 10 values are used in each dimension. The gradient threshold is varied on a logarithmic scale from 20Hz/second to 500 Hz/second, and the deletion threshold is varied linearly from 0.025 seconds to 0.475 seconds. Using each set of thresholds, the system is run over a set of data, the transcriptions produced are compared with the hand labelled versions, and the recognition score for that set of thresholds is recorded. After all possibilities have been tried, the set of thresholds giving the best recognition score is chosen as being the optimal set.

This technique can be used to train on any amount of data, but 10 utterances are sufficient to ensure safe training as recognition scores do not significantly improve with more training data. In all data sets, a deletion threshold of 0.075 or 0.125 seconds was chosen as best[5]. The gradient thresholds varied more between speakers, from 70Hz/second to 120Hz/second. These thresholds do not give an indication of how steep the rise and fall elements actually are; rather, they are the optimal thresholds that distinguish legitimate rises and falls from connection elements and obstruent perturbations.

6.3 Analysis Results

Table 3 shows the results of the automatic labeller for the six speakers. The overall accuracy rate in the high 70s leaves room for improvement, but considering the fact that the system is working in an unconstrained fashion with no top-down processing, these results are promising. The errors were examined to discover their source, and the results from this study showed that the overall picture is much better than the above results might lead one to believe.

Four sources of error were identified. These were:

F_0 errors A small number of the errors were due to the PDA making a mistake such as pitch doubling. This causes a sudden jump in the F_0 contour which can be mistakenly interpreted as a rise or fall element. As Hess (1983) (page 66) notes, these gross F_0 errors can often be compensated for when hand labelling, as the eye is able ignore the erroneous values and detect the underlying pattern.

Obstruent Perturbations The F_0 post-processor and deletion module did not account for all obstruent perturbations and a number of errors arose from the labeller mistaking glitches or spikes for small rise or fall elements. The insertion errors, where a perturbation is mistaken for a rise or fall, can easily be eliminated by increasing the gradient thresholds, but this has the effect of causing deletion errors as small genuine elements are not detected. So far the system has worked using F_0 as input alone: it might be necessary in future implementations to use additional information. For instance, a phonetic segmentation would give information on where obstruent perturbations occur, and heavier smoothing could be used in these areas. If better post-processing

[5] 0.075s corresponds a single 50 ms frame being deleted (50ms < 75 ms) and 0.125s corresponds to two 50 ms frames being deleted (2 x 50ms < 0.125).

was used fewer perturbations would be classed as rises or falls, and the gradient thresholds could be lowered so as to accept more of the genuine rises and falls.

Data set A had a considerably higher recognition rate than the other sets and this was mainly due to the speech in set A being fully voiced and being freer from obstruent perturbations than the other sets.

Algorithmic Problems Some errors arose from straightforward mistakes by the labelling algorithm arising from phenomena not foreseen in the initial design. A common mistake was for downstepping pitch accents on successive syllables to be labelled as a single large fall. Here the system would detect a long falling section of F_0 and assume this to be a single accent. A simple modification to the optimal matching module allowing more than one fall shape to be fitted to elements that have been labelled as falls by the broad class labeller should help solve this problem.

Labelling Problems A small number of errors arose from situations where is was not clear that the hand labelling was correct. Nearly always these cases involved small phrase initial boundary rises or small pre-nuclear pitch accents. The arbitrary nature of the hand labelling in these cases was not such much of an inherent problem in the RFC model, as in any system it is often difficult to decide whether certain stressed syllables should be marked as having pitch accents or not.

The majority of errors arose from problems with small pitch accents or phrase initial boundary rises. However, it is a general feature of the intonational system of English that perceptually important accents are bigger than those which are not[6], and therefore it is the case that the system recognised important accents best. To confirm this, an additional test was performed which measured the accuracy of the labeller on rises and falls which were part of nuclear accents. The results given in table 4 clearly demonstrate that the automatic labeller is very successful at finding and classifying nuclear accents.

Graphs of F_0 synthesis and Analysis

Figures 3 and 4 show F_0 contours from two utterances in set A. In each figure, graph (a) is the original F_0 contour as produced by the post-processing module. Graphs (b) and (c) show the original contour, the labels from an RFC description and a synthesized contour from this RFC description. Graph (b) shows a hand labelled RFC description, and graph (c) shows an automatically labelled RFC description.

These figures demonstrate a number of points. By superimposing the synthesized F_0 contours on the originals it is possible to gain a subjective impression of the model's synthesis accuracy. While slight differences between the F_0 contours can be detected, it is clear that the synthesized versions are close to the originals, supporting the objective evidence in table 2.

The (c) graphs show some typical errors from the automatic labeller. The second pitch accent of the first phrase in figure 3 is mislabelled as the automatic system failed to recognise the small fall element of this accent. In figure 4, an insertion error occurs as an obstruent perturbation was mislabelled as a fall element. These graphs support the evidence in table 3 which shows that the automatic labeller labels large pitch accents more accurately than smaller ones.

6.4 Other Automatic Analysis Systems

A number of automatic intonation analysis systems have recently been proposed, but it is difficult to make comparisons owing to the different nature of the tasks attempted by these systems. Jensen et al. (1993) describe a system that recognises O'Connor and Arnold tune types from F_0 contours. They report a 71% accuracy in classifying nuclear accents, which is worse than the results reported in table 4, but as they are attempting a full phonological description, their basic task is much harder. Geoffrois (1993) describes a recognition scheme for Japanese speech using the Fujisaki model where he correctly recognises 91% of accent commands. Although the step and impulse functions of the Fujisaki model are roughly equivalent in terms of level to the rises and falls of the RFC model, direct comparison is again difficult due to the fundamentally different nature of English and Japanese intonation.

[6] Many factors such as pitch range and declination need to be taken into account, but it does seem generally true that accents which are important, e.g. nuclear accents are consistently larger than those which are not.

7 Discussion

7.1 Flexibility and Variance in the RFC model

It is simple to devise a model which can synthesize F_0 contours with a high accuracy: any sort of unconstrained target system can do this. What is much more difficult is to design a model with high synthesis accuracy that generates these F_0 contours from a *linguistically useful description*. It should be clear that the RFC descriptions *are* linguistically useful due to the well-formedness conditions given in section 2.2. These conditions state that rise and fall elements may only be used to model phonologically significant events, and phonologically significant events can only be described using rises and falls. Therefore all phonological events are readily detectable in an RFC description.

The only debatable point is whether the flexibility in the RFC description is justified. The free parameters in the model are the durations and amplitudes of the rise and fall elements. A typical pitch accent with a rise and fall element thus needs four parameters. In our data the amount of variance in pitch accents is considerable. A previous study of data set A showed that rise element amplitudes vary from 10Hz to 96Hz and fall element amplitudes vary from 11Hz to 140Hz (Taylor, 1992). This is not controversial as most systems have some way to gradiently scale accent height (eg (Liberman and Pierrehumbert, 1984). However the RFC model also allows variability in the slope gradients of the rise and fall elements.

Table 5 shows statistics derived from the full set of hand RFC labels for speaker C. The large variance of the amplitudes, durations and gradients should make it clear that these parameters do indeed need to be flexible to account for the data. Any model failing to acknowledge this variance will have considerably worse synthesis accuracy than the RFC model.

Informal investigation into the causes of the variance shows relationships between element duration, gradience and the sonority of syllables, ie. syllables with short voiced regions have shorter durations and steeper gradients than those with long voiced regions. Nuclear accents occur earlier with respect to syllable boundaries than pre-nuclear accents, which is in line with the more thorough study of Silverman and Pierrehumbert (1990). The wide amount of variance is therefore probably due to differences in the sonority of the syllables, the position of accents in the phrase and the phonological class of the pitch accent. Thus the flexibility in the model is required if it is to synthesize F_0 contours with high accuracy. The strength of the RFC model lies in its ability to do this while still making the RFC descriptions easily amenable to phonological analysis.

7.2 Comparison with other Models

Fujisaki

Fujisaki's model of Japanese intonation uses two critically damped second order filters to generate F_0 contours. The *phrase component* uses impulses as input and models long term phenomena such as downdrift and resets. The *accent component* uses step functions as input and models pitch accents. Separate time constants control the rate of rise and fall of each component, and in the classical definition of the model, these constants are invariant for a speaker.

The Fujisaki accent component only requires two numbers, amplitude and duration, to model pitch accents compared with the four mentioned above for the RFC model. Thus the Fujisaki model is more constrained than the RFC model. However, the accent component cannot synthesize the pitch accents in our data with the accuracy of the RFC model. Amendments can be made, such as allowing a negative step in the later half to account for downstepping accents, but this would add two extra parameters (the amplitude and the position of the polarity change). Furthermore, the Fujisaki model predicts that pitch accent gradience is invariant for a speaker, and as table 5 shows, the gradient of rises and falls vary considerably for a speaker. Therefore the Fujisaki model would have to allow the time constant to vary also, adding even more parameters.

The phrase component is problematic as it cannot model long continuously rising sections of F_0 contour, such as commonly observed after L* accents (as seen in figure 2 of Beckman and Pierrehumbert (1986)), or in the pre-nuclear position in the surprise redundancy contour (as seen figure 11 in Ladd (1983)). Figure 4 also demonstrates this effect. We found it difficult to postulate any amendment to the model which would account for such phenomena.

It must be noted that the intonational system of Japanese is quite different from that of English, and in particular

it has a much smaller inventory of pitch accents (Beckman and Pierrehumbert (1986), like Fujisaki, use only one). Thus it is to expected that a model developed with the Japanese language in mind would be restricted in pitch accent shape.

Dutch

In the Dutch school (t'Hart and Cohen, 1973), (t'Hart and Collier, 1975), (Willems et al., 1988), F_0 contours are analysed in two stages. The *close-copy stylization* describes the F_0 contour in terms of a number of straight lines. These close-copy stylizations are claimed to be perceptually indistinguishable from the original F_0 contours, as demonstrated by re-synthesis tests (Willems et al., 1988). This is essentially a data reduction exercise. The second stage is termed *standardization* whereby the close-copy contour is further coded into a series of standard rise and fall patterns.

Although the close-copy process may produce F_0 contours with a synthesis accuracy near that of the RFC model, the standardization process makes the Dutch model more like the prototypical systems mentioned in the introduction. Therefore the aim is not to accurately synthesize any F_0 contour, but to discover a set of F_0 patterns which are within the "linguistic tolerance" of a listener (Willems et al., 1988).

Hirst

The RFC model is similar in some ways to the model proposed by Hirst (1992) where an attempt is made to derive the phonological description of an utterance from its F_0 contour. The first stage of his system uses a spline fitting algorithm that reduces an F_0 to a number of target points. Later these target points are classified in terms of a surface phonological description. Although we are not aware of any exact performance figures for Hirst's system, it would be likely that his spline model's synthesis accuracy should be least as accurate as the RFC model, as his target points are not as constrained as the RFC model. The RFC model lies somewhere between Hirst's spline level and the surface phonological level in that the location of the pitch accents and the boundary rises are explicitly marked in the RFC description whereas the spline description needs further processing to extract this information.

7.3 Practical Applications

The RFC model forms the basis of the intonation component in the speech synthesis system being developed at ATR. In this system, intonational tune is described by a system of intonational elements (H, L etc) and features (delayed, downstep etc), which is similar to the intonational tune phonology of Ladd (1983). F_0 contours from the ATR-CMU database (including data sets C to F) have been labelled using the RFC model and the tune phonology. From the RFC descriptions of a speaker's utterances, it has been possible to collect statistics on the amplitudes and timing characteristics of pitch accents. These statistics are used in the intonation component of the speech synthesis system to ensure that the synthesized intonation has a good likeness to the original speaker. It is much easier to derive these statistics from an RFC description than directly from an F_0 contour, as the parts we are most interested in (e.g. pitch accents) are explicitly marked in an RFC description in a regular manner. A description of the tune phonology system and its relation to Ladd's is given in Taylor (1992), and a description of the intonational component of the speech synthesis system is given in Taylor (1993b).

We are currently working on integrating the automatic labeller into the ATR speech recognition system. A system already exists which uses the RFC description from the automatic labeller to derive the intonational tune of the utterance (Taylor, 1993a). Work is underway to use this tune description to help derive the speech act (question/statement/greeting etc) of the utterance.

The synthesis of F_0 contours from an RFC description is computationally trivial, while the automatic labeller takes about 0.3 seconds of processing time for every 1.0 second of speech. Faster performance figures should be possible as no speed optimisation has been attempted on these programs.

7.4 Conclusion

As far as extending the model to produce a higher level analysis based on an RFC description is concerned, two points can be made. Firstly, due to the fact we can reconstruct a contour very similar to the original one, no information present in the original F_0 has been lost; rather it has been converted to a form more amenable to further

analysis. Secondly, in an RFC description, pitch accents and boundary rises are identified, and further analysis need only concentrate on classifying the rise and fall descriptions into separate phonological classes of pitch accent and boundary tune.

In conclusion, describing intonation with rise, fall and connection elements is very useful in that this description is relevant to both the phonological and acoustic descriptions of intonation. Pitch accent and other intonational tune information can be derived from an RFC description, and F_0 contours can be accurately synthesized from the RFC description.

References

Anderson, M. D., Pierrehumbert, J. B., and Liberman, M. Y. (1984). Synthesis by rule of English intonation patterns. In *International Conference on Speech and Signal Processing*. IEEE.

Bagshaw, P. C., Hiller, S. M., and Jack, M. A. (1993). Enhanced pitch tracking and the processing of f0 contours for computer aided intonation teaching. In *Proc. Eurospeech '93, Berlin.*

Beckman, M. E. and Pierrehumbert, J. B. (1986). Intonational structure in Japanese and English. *Phonology Yearbook 3*, pages 255–309.

Fujisaki, H. and Kawai, H. (1988). Realization of linguistic information in the voice fundamental frquency contour of the spoken Japanese. In *International Conference on Speech and Signal Processing*. IEEE.

Geoffrois, E. (1993). A pitch contour analysis guided by prosodic event detection. In *Proc. Eurospeech '93, Berlin.*

Grice, M. L. (1992). *The Intonation of interrogation in Palermo Italian; implications for intonation theory.* PhD thesis, University College London.

Halliday, M. A. K. (1967). *Intonation and Grammar in British English.* Mouton.

Hess, W. (1983). *Pitch Determination of Speech Signals.* Springer-Verlag.

Hirst, D. (1992). Prediction of prosody: An overview. In Bailey, G. and Benoit, C., editors, *Talking Machines.* North Holland.

Isard, S. D. and Pearson, M. (1988). A repertoire of British English contours for speech synthesis. In *SPEECH '88, 7th FASE Symposium.* FASE.

Jensen, U., Moore, R. K., Dalsgaard, P., and Lindberg, B. (1993). Modelling of intonation contours at the sentence level using chmms and the 1961 O'Connor and Arnold scheme. In *Proc. Eurospeech '93, Berlin.*

Ladd, D. R. (1983). Phonological features of intonation peaks. *Language*, 59:721–759.

Ladd, D. R. (1987). A model of intonational phonology for use with speech synthesis by rule. In *European Conference on Speech Technology.* ESCA.

Ladd, D. R. (1988). Declination reset and the hierarchical organization of utterances. *Journal of the Acoustical Society of America*, 84(2):530–544.

Liberman, M. and Pierrehumbert, J. (1984). Intonational invariance under changes in pitch range and length. In Aronoff, M. and Oehrle, R. T., editors, *Language Sound Structure.* MIT Press.

Medan, Y., Yair, E., and Chazan, D. (1991). Super resolution pitch determination of speech signals. *IEEE Trans. Signal Processing*, 39:40–48.

O'Connor, J. D. and Arnold, G. F. (1973). *Intonation of Colloquial English.* Longman, 2 edition.

Pierrehumbert, J. B. (1980). *The Phonology and Phonetics of English Intonation.* PhD thesis, MIT. Published by Indiana University Linguistics Club.

Rabiner, L. R., Cheng, M. J., Rosenberg, A. E., and McGonegal, C. A. (1976). A comparative study of several pitch detection algorithms. *IEEE Trans.*, ASSP-24(5):399–418.

Sagisaka, Y. and Kaiki, N. (1992). Optimization of intonation control using statistical f0 resetting characteristics. In *International Conference on Speech and Signal Processing*. IEEE.

Silverman, K. and Pierrehumbert, J. B. (1990). The timing of prenuclear high accents in English. In Kingston, J. and Beckman, M. E., editors, *Papers in Laboratory Phonology 1*. Cambridge University Press.

Taylor, P. A. (1992). *A Phonetic Model of English Intonation*. PhD thesis, University of Edinburgh.

Taylor, P. A. (1993a). Automatic recognition of intonation from F_0 contours using the rise/fall/connection model. In *Proc. Eurospeech '93, Berlin*.

Taylor, P. A. (1993b). Synthesizing intonation using the RFC model. In *Proc. ESCA Workshop on Prosody, Lund, Sweden*.

ten Bosch, L. (1993). On the automatic classification of pitch movements. In *Proc. Eurospeech '93, Berlin*.

t'Hart, J. and Cohen, A. (1973). Intonation by rule: a perceptual quest. *Journal of Phonetics*, 1:309–327.

t'Hart, J. and Collier, R. (1975). Integrating different levels of intonation analysis. *Journal of Phonetics*, 3:235–255.

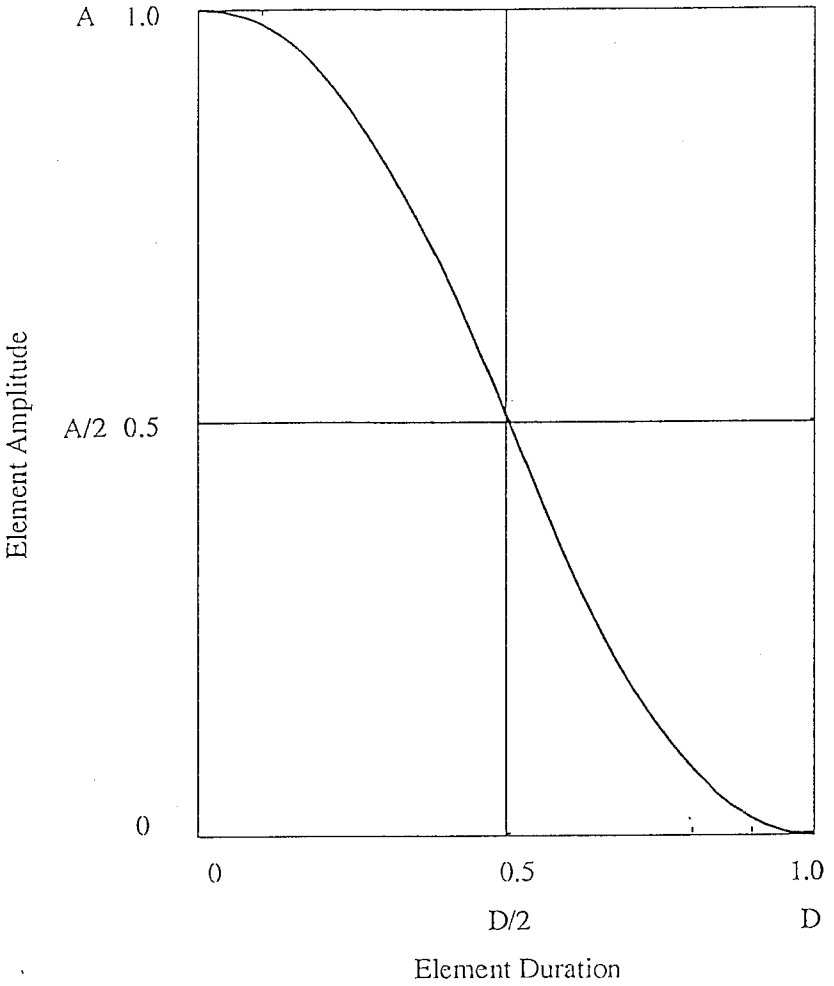
Traber, C. (1990). F_0 generation with a database of natural F_0 patterns and with a neural network. In *Proceedings of the ESCA Workshop on Speech Synthesis*, pages 141–144.

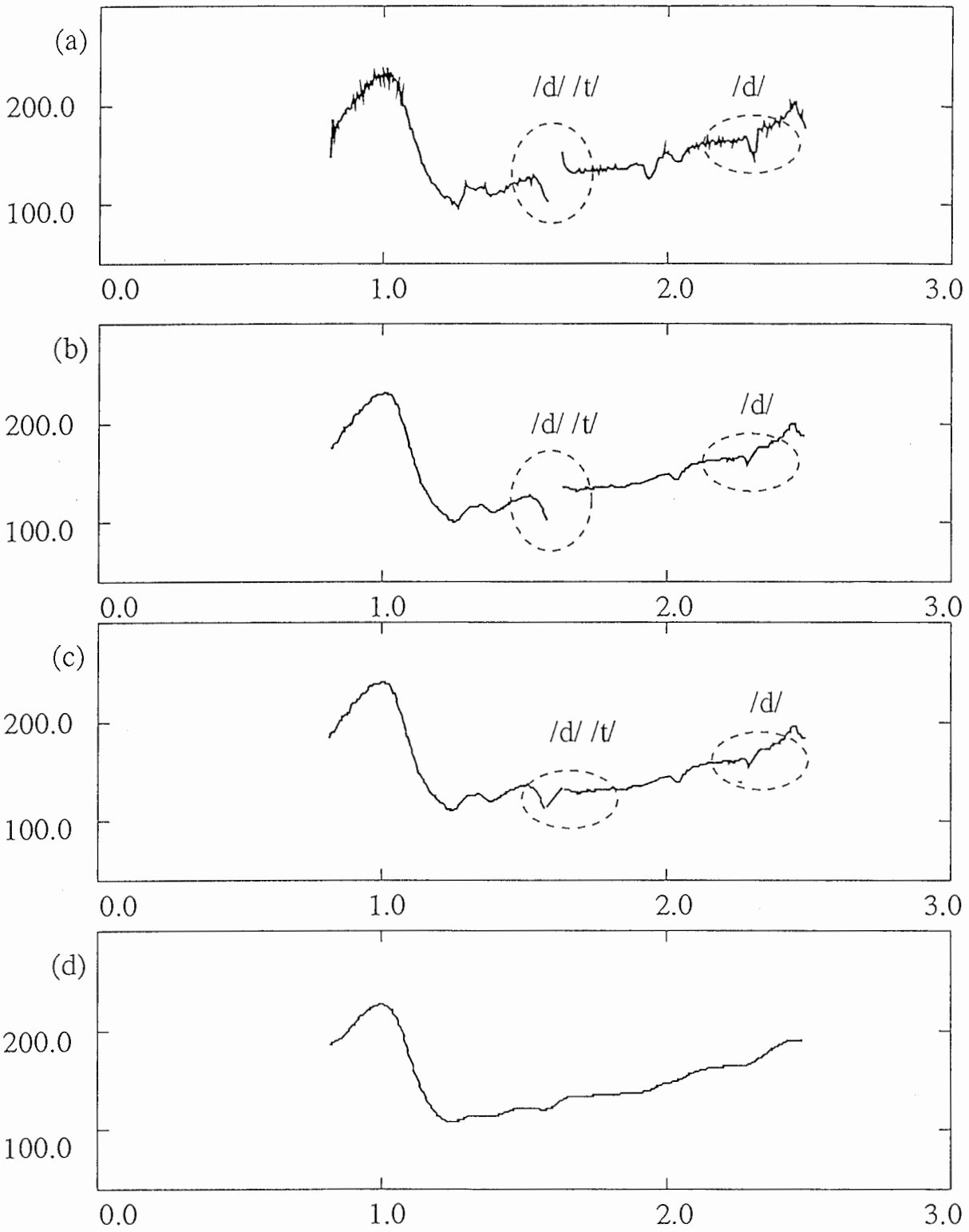
Vonwiller, J. P., King, R. W., Stevens, K., and Latimer, C. R. (1990). Comprehension of prosody in synthesized speech. In *SST-90, Third International Australian Conference in Speech Science and Technology*.

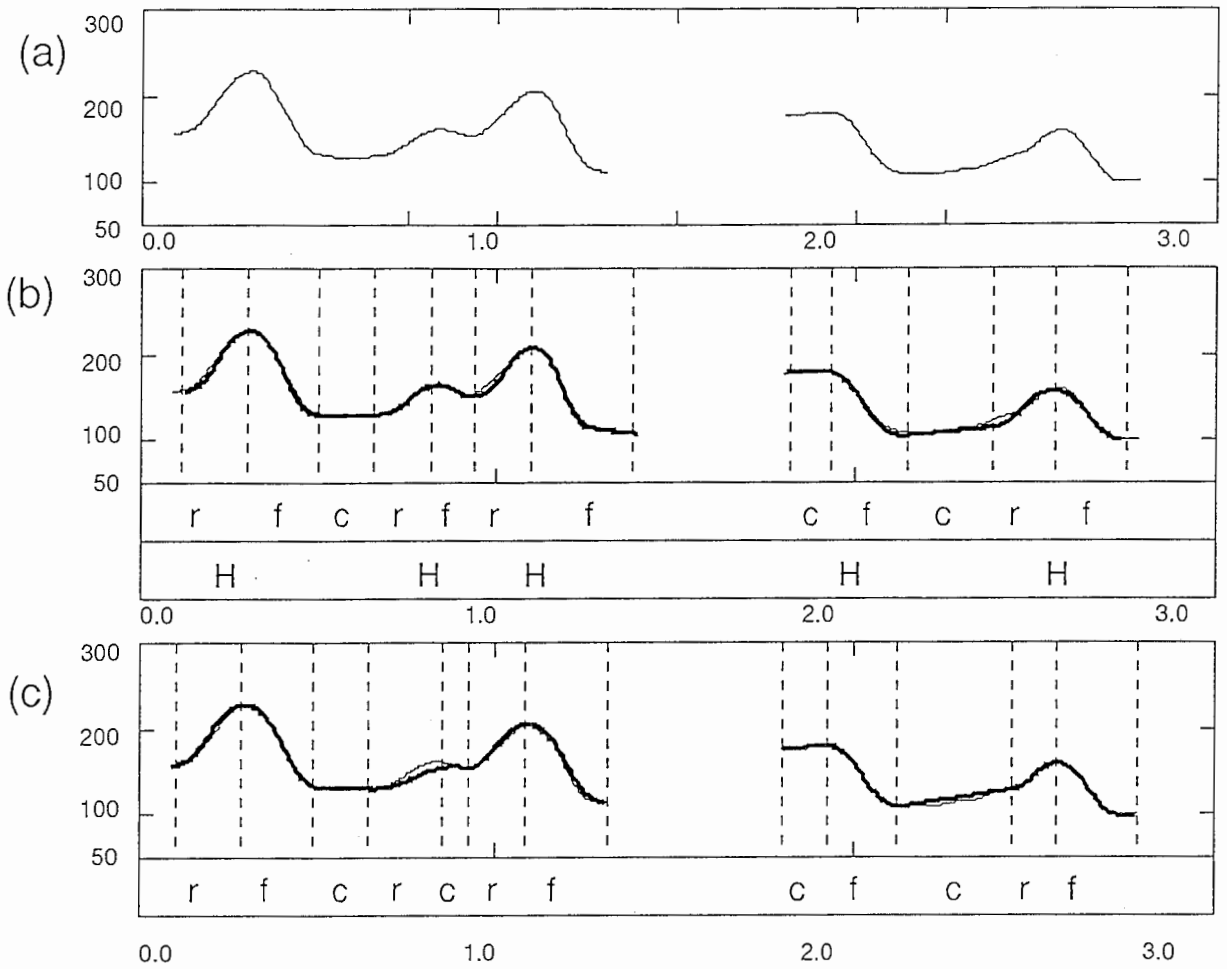
Willems, N. J. (1983). A model of standard English intonation patterns. *IPO annual Progress Report*.

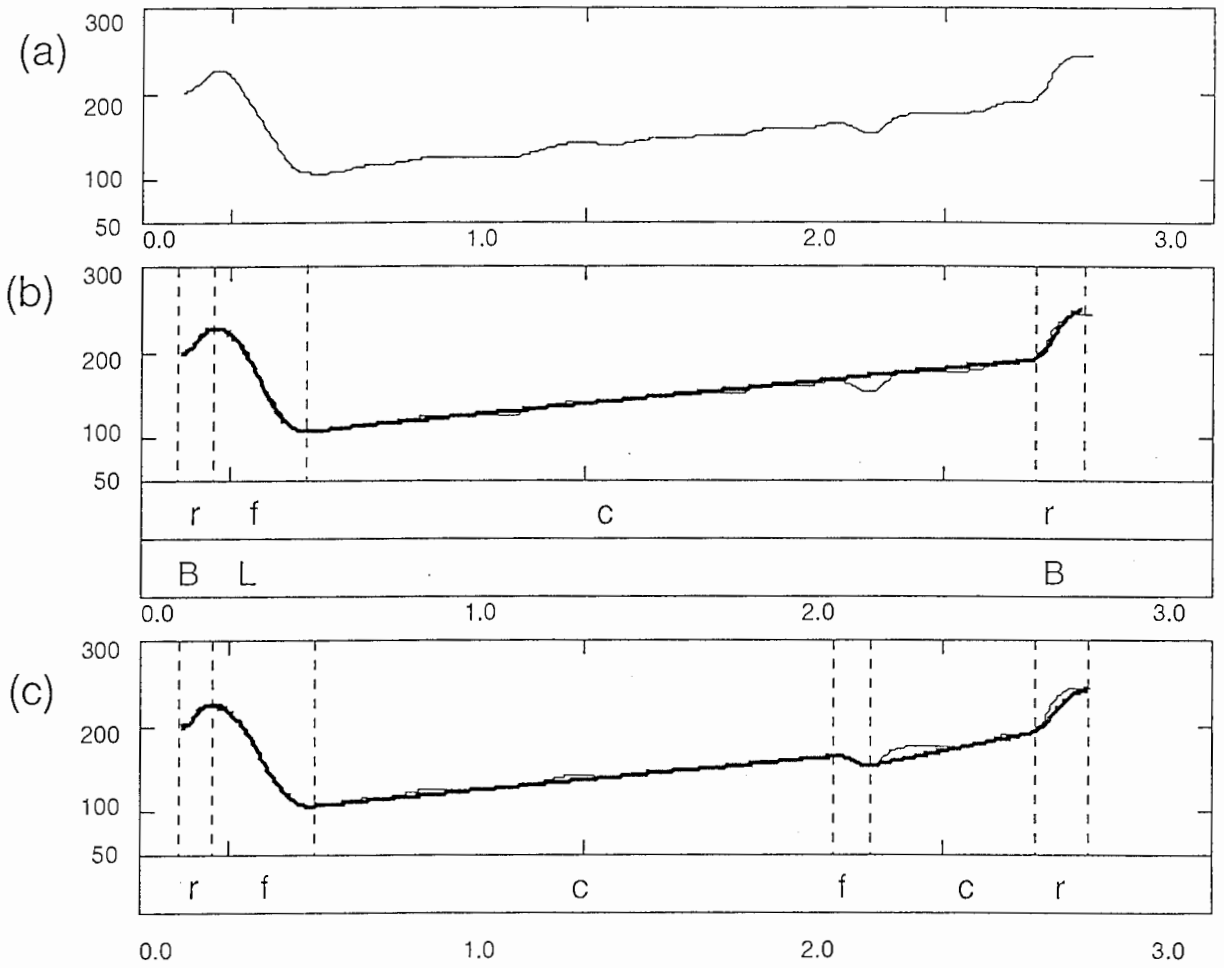
Willems, N. J., Collier, R., and t'Hart, J. (1988). A synthesis scheme for British English intonation. *Journal of the Acoustical Society of America*, 84(4):1250–1261.

Wodd, C. A. (1992). ATR Interpreting Telephony - Carnegie Mellon University conference registration task. Technical report, ATR Interpreting Telecommunications Labs. Ed. David Rainton.


A 1.0
A/2 0.5
0
Element Amplitude
0
0.5
1.0
D/2
D
Element Duration


(a)
200.0
100.0
/d/ /t/
/d/
0.0
1.0
2.0
3.0
(b)
200.0
100.0
/d/ /t/
/d/
0.0
1.0
2.0
3.0
(c)
200.0
100.0
/d/ /t/
/d/
0.0
1.0
2.0
3.0
(d)
200.0
100.0
0.0
1.0
2.0
3.0


(a)
300
200
100
50
0.0
1.0
2.0
3.0
(b)
300
200
100
50
r f c r f r f
c f c r f
H H H H H
0.0
1.0
2.0
3.0
(c)
300
200
100
50
r f c r c r f
c f c r f
0.0
1.0
2.0
3.0


(a)
300
200
100
50
0.0
1.0
2.0
3.0
(b)
300
200
100
50
r
f
c
r
B
L
B
0.0
1.0
2.0
3.0
(c)
300
200
100
50
r
f
c
f
c
r
0.0
1.0
2.0
3.0

Type	Duration (seconds)	Amplitude (Hz)
rise	0.187	70
fall	0.187	-97
conn	0.175	0
rise	0.165	34
fall	0.100	-14
rise	0.171	57
fall	0.159	-93
conn	0.135	-7
silence	0.405	73
conn	0.105	0
fall	0.225	-76
conn	0.240	10
rise	0.175	43
fall	0.191	-57

Data set	Number of utterances	From hand labels	From automatic labels
A	64	4.9 Hz	4.7 Hz
B	45	7.3 Hz	5.4 Hz
C	55	3.6 Hz	4.2 Hz
D	19	4.1 Hz	4.3 Hz
E	21	3.7 Hz	3.8 Hz
F	17	4.9 Hz	3.9 Hz

Data set	Number of utterances	Numbers of elements	% of rise and falls correct
A	64	352	92
B	45	589	86
C	55	332	75
D	19	125	72
E	21	138	74
F	17	109	72

Data set	Number of utterances	Number of nuclear accents	% Correct
A	64	136	98.5
B	45	156	95.4
C	55	139	97.6
D	19	34	94.1
E	21	39	94.8
F	17	29	96.5

Parameter	Mean	Standard Deviation
rise gradient	18 st/sec	15
fall gradient	-26 st/sec	15
rise duration	157 ms	92
fall duration	165 ms	82
rise amplitude	2.57 st	1.93
fall amplitude	-4.18 st	2.49

Figure 1

The quadratic monomial function. The plot is shown in the x and y space, and also with the axes marked for duration and F_0 amplitude.

Figure 2

Contour (a) is the normal output of the PDA. Contour (b) shows the result after 15 point smoothing. Contour (c) shows the interpolation through the unvoiced regions and contour (d) shows the final output of the modified PDA.

Figure 3

Graphs of the utterance "The large window stays closed: the small one you can open". Graph (a) shows the original F_0 contour. In graphs (b) and (c) the original F_0 contour is shown by the thin line and the synthesized F_0 contour is show by the thick line. Rise, fall and connection elements are labelled "r","f" and "c". In the second label box of the graph (b), the "H" symbol indicates the presence of a high or peak pitch accent.

Figure 4

Graphs of the utterance "Must you use so many large unusual words when you argue". The same labelling conventions apply as for figure 3 with the addition that "L" indicates a low or valley accent and "B" indicates a boundary rise.

Note: I recommend that figures 3 and 4 should be printed using the full width of the page.

Table 1

Example of RFC description. These labels were derived using the criteria explained in section 6.1

Table 2

Average distances in Hertz between synthesized and original F_0 contours for hand and automatic labels.

Table 3

Percentage recogbition scores for the automatic labeller on the six sets of data.

Table 4

Percentage recognition scores for the automatic labeller on the nuclear accents of the six sets of data.

Table 5

Mean and standard deviations for hand labelled element parameters for speaker C. "st" stands for semi-tones, "sec" for seconds and "ms" for milli-seconds