ARIPUC 16, 1982, p. 117-126

SPEECH SYNTHESIS AT THE INSTITUTE OF PHONETICS

PETER HOLTSE

This paper gives a brief description of the more important research involving speech synthesis at the Institute of Phonetics since the Institute was founded in 1966. Further, it provides a status report for ongoing research in the shape of a more detailed account of our current activities and future plans.

I. HISTORY

1. HARDWARE SYNTHESIZER

The first workable synthesizer was constructed at the Institute in the late 1960'es when Jørgen Rischel built an electronic synthesizer. At first ambitions were low, and a machine capable of producing isolated vowels was developed (Rischel, 1966). It was, however, soon followed by a very sophisticated dynamically controlled synthesizer capable of producing connected speech of a quality only limited by the skill of the operator and the time available.

The synthesizer and its control panel is described in detail in Rischel and Lystlund (1972). As documented here it is a very versatile instrument for perceptual research since it is capable of producing almost any conceivable speech sound (and quite a few non-speech sounds). However, the complexity of its control structure (it is a parallel formant synthesizer with twenty variable parameters) makes the task of producing anything like connected speech forbidding. Yet, the analogue control panel, with its facilities for interactive modification to any part of the speech signal and with immediate auditory feedback, is almost unrivalled for sequences of one or two syllables.

Of course, the synthesizer has its drawbacks. Thus, one source of continuous irritation has always been that the settings of the knobs of the control panel can only be adjusted by hand, i.e. the information on the panel cannot be saved for later use in any way. Furthermore, since the whole system is constructed in analogue technique it needs extensive calibration if exact control over, for instance formant frequencies, is needed.

The natural answer to these deficits was to simulate the actions of the control panel on the computer. A program doing this was operational on the Lab PDP-8 computer in 1977 (described in an internal report: Holtse (1978)). This program added visual feedback to the system since it would display the settings of the control panel on a graphic terminal. However, somehow the computer control program never became as popular as the old analogue control panel. Maybe because the immediate access to all parameters provided by the knobs of the control panel had to be substituted by keyboard commands, which many people find less attractive.

As might have been expected from a hybrid system, the problem of analogue drift was actually enhanced by the added computer control. When people type "335 Hz", they like to think that they get 335 Hz and not "something between three and four hundred". Although it was possible, in principle at least, to calibrate the simulated control panel quite accurately, this remained a time consuming task. And some of the problems, for instance that of calibrating formant amplitudes, have never really been solved.

In spite of obvious drawbacks the synthesizer has been used - either with its original analogue control panel or under computer control - in a number of perceptual studies, mostly dealing with vowel perception (Holtse (1973), Reinholt Petersen (1974 and 1976), and Rosenvold (1981)). Although largely undocumented its value as a pedagogical tool has been considerable thanks to the possibilities of interactive work.

2. LIBRARY OF SPEECH SOUNDS

Quite early, the idea of having a library of standard settings for the more common speech sounds occurred. The need for such a library was the more obvious since the complex control structure of the synthesizer made the proper working of the machine very much a job for specialists.

At the same time the Telecommunications Research Laboratories of Copenhagen became interested in the possibilities of producing synthetic speech by rule. This problem had not been given much thought at the Institute of Phonetics, where most people were interested in speech perception in a very general way. However, it was felt that there was enough common interest to initiate a small research project in synthesis by rule, and a first system was developed. It is described in Holtse (1974).

The system consisted largely of a library of formant coded speech sounds and a set of interpolation routines. The system would accept input in phonetic transcription, and it produced intelligible Danish words of two or three syllables. However, control of prosodic information was extremely primitive, and longer sequences needed hand editing of pitch and duration to become intelligible.

A more sophisticated prosodic control system was not developed. Mainly because at that time the phonetic description of Danish pronunciation was insufficient to allow meaningful rules concerning pitch and duration to be formulated.

3. SIMULATION OF PHONOLOGICAL RULES

Although not strictly speech synthesis the work on computer testing of phonological rules documented in Basbøll and Kristensen (1974 and 1975) should be mentioned in this connection. Their computer program would accept a sort of underlying phonological structure as input and produce a rather detailed phonetic transcription as output. (It would actually produce several different transcriptions since the authors were particularly interested in the problems of alternative pronunciations in different styles of speech.)

As the transformation rules of the system were developed, the "underlying structure" gradually approached standard Danish orthography. And it was realized that it might be possible to produce a Danish text-to-speech system - even if the accepted view among phoneticians had been that the pronunciation of Danish text cannot be predicted without access to the meaning of the text.

At that time the plan was to use the output of the rule testing program as input to the synthesis program mentioned in the previous paragraph. This plan was, in fact, never carried out since both projects were discontinued before it could be tried.

II. CURRENT ACTIVITIES

Towards the end of the 1970'es several people expressed an interest in combining the research in Danish phonology with the work on analysis and synthesis of Danish speech. In particular, the idea of producing a complete text-to-speech system for Danish was beginning to mature. Till then the development of such a system had been of only limited interest to phoneticians and phonologists. But the thought of combining the research efforts of several linguistic-phonetic areas was appealing from both a theoretical and a practical point of view.

A serious difficulty for the project was that it needed considerably more computer power than could be made available at the Institute of Phonetics, where the economic crisis was beginning to make itself noticeable. However, a Danish synthesis-by-rule system would be of practical value to many people outside the academic community. And the newly formed project group received sufficient support from, in particular, the telecommunications world, to obtain a grant from the Thomas B. Thrige's Foundation and the Tuborg Foundation for the purchase of the necessary equipment. (The computer system is described elsewhere in this issue.)

Since then, our activities have been centered in two areas: 1) Development of suitable Danish text-to-phonetic transcription algorithms, and 2) Development of support tools for further research in speech synthesis. The first part, the orthographic/phonological research, is described in the paper by Peter Molbæk Hansen (this issue), while the rest of the present paper is devoted to a description of the planned support system.

A. SURVEY OF SYNTHESIS SUPPORT FACILITIES

Figure 1 shows the logical connections between various parts of the support system. The figure shows four main components: At the top (labelled "A") is traditional phonetic/phonological research.

The models and descriptions of Danish being developed here are, as far as possible, coded in a special formal language, in its syntax very like the notation known from "The Sound Pattern of English" (Chomsky and Halle (1968)). The rules and descriptions are then processed on the computer by a Rule Compiler (shown in section "B" of figure 1) to produce the actual speech synthesis program (section "C").

The idea of having a special speech synthesis computer language is not our own. It was first described by Carlson and Granström (1974). Our own compiler is in fact modelled closely on the Swedish compiler, and we have profited immensely from discussions with Rolf Carlson and Björn Granström.

The text-to-speech conversion program produced by the Rule Compiler is planned to take a string of ASCII text in normal Danish orthography as input and from this produce a set of time varying control parameters which, when fed into the appropriate speech synthesizer will eventually produce the corresponding spoken word.

Alternatively, the control parameters generated by the text-to-speech program may be stored on disk files and manipulated by the Phonetic Debugging System shown in section "D" of figure 1. The purpose of the parameter editor and its associated display programs is to put as many as possible of the facilities of the old hardware control panel previously mentioned at the disposal of the phoneticians (and hopefully provide a few new ones).

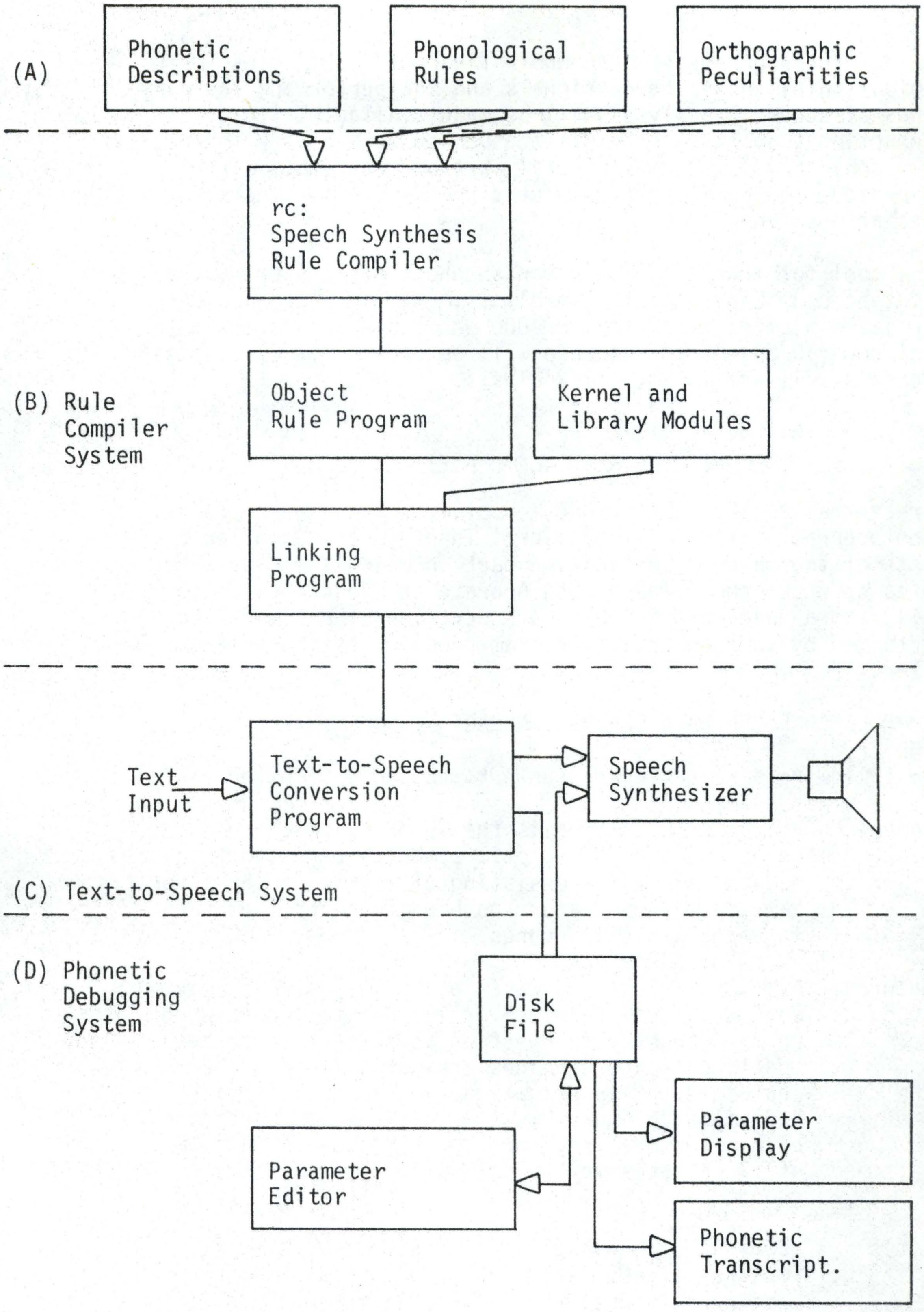

Figure 1

Block diagram of Synthesis Support System

Besides the programs etc. shown in figure 1, facilities for digital sampling of speech signals and for performing various kinds of acoustic analyses will be made available. In this connection a database of digitized Danish speech is planned. This could be useful as a general sort of reference, although we believe the majority of phonetic problems will need special data in any case.

As a tool for the analysis of Danish phonology and orthography a database of Danish texts, wordlists, etc. is planned. Currently, a dictionary of some 50.000 words has been read in (Holmboe (1978)). The database will be accessible for statistical as well as syntactical analysis.

B. THE RULE COMPILER

Our central tool in the production of a text-to-speech conversion program is the Rule Compiler. Input to the Compiler consists of two parts: A definition part in which the data structures on which the rules are to operate are described in detail, and a Rule Part in which are described the actions to be performed by the conversion program produced by the Rule Compiler.

1. RULE COMPILER DATA DEFINITION PART

The following structures are known to the Rule Compiler:

Segments The basic data type is the segment, which roughly corresponds to a phonetic symbol. Each segment is a data structure consisting of a Name, a Character Name, a unique Feature combination, and from zero to three acoustic chunks.

Features Features are binary descriptive labels which may assume values of "plus" or "minus" to denote whether the property in question is present or absent. Undefined feature values are not allowed. The Feature "seg" is predefined in the Rule Compiler. Additional Features must be defined as belonging either to the [+seg] or [-seg] group. A maximum of 31 Features may be defined for each of the two groups.

Labels Labels are combinations of one or more specific Feature values. They are merely notational conveniences. Thus, the Label "V" for Vowel could reasonably be used as shorthand for the Feature combination [-cons, +voc, +syll].

Chunks Chunks are the acoustic descriptions of the individual segments. Typically a short vowel will consist of only one chunk, while a plosive consonant may contain two or three.

Each chunk consists of a set of Variables and a set of Parameter structures.

Variables are integer values used to describe the chunk. Two Variables are predefined in the Rule Compiler: "dur" and "rank". "dur" defines the physical duration of the chunk in milliseconds. "rank" defines which chunk should dominate over the neighbours when the chunks are combined to make connected speech.

Parameters are integer values corresponding to the control parameters of the speech synthesizer on which the output of the rules will eventually be applied. For each Parameter a data structure is defined, consisting of a Target value, an Internal Transition time, and an External Transition time. The two transition times describe the respective points in time when the Parameter should either leave one Target or reach the next Target.

Typical Parameters would be "f1": frequency of first formant, or "f0": pitch of voiced excitation.

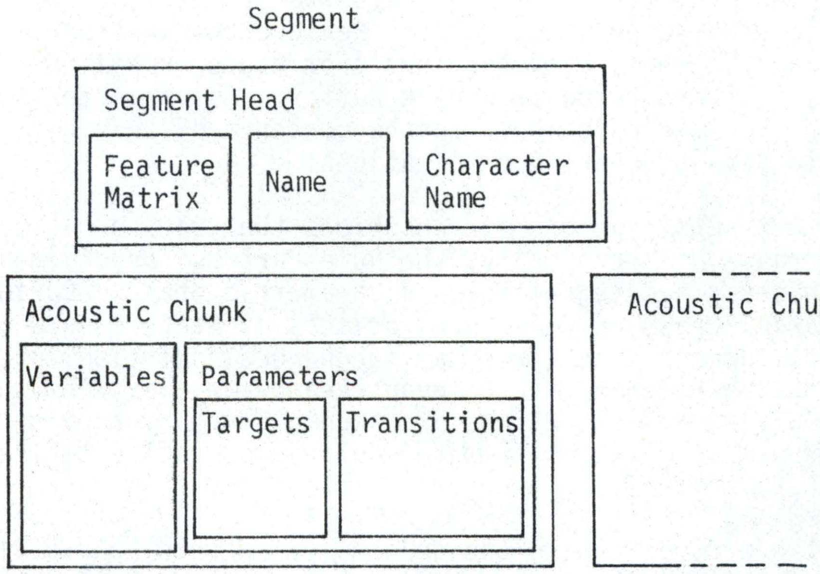

Figure 2

Structure of a Segment

Apart from the Segment data type, the Rule Compiler recognizes "integer" and "real" global variables. These are variables which do not belong to particular Segments and may be used for computations or temporary storage within the rules.

All data structures must be defined prior to their use in any rule.

As will be noted all data structures are segmental, i.e. words and syllables are indicated by special delimiting segments. This procedure is according to tradition. However, the notation of rules applying to higher levels than the segment tends to become extremely complicated. We are therefore presently considering introducing structures like syllables and words in the Rule Compiler.

2. RULE COMPILER RULE DESCRIPTION PART

The object of the Rule Compiler is, as previously mentioned, to produce a computer program which will convert a string of ASCII characteris representing Danish orthography into time varying control parameters needed by a speech synthesizer. This is accomplished as follows.

The conversion program reads a string of characters up to (and including) the first sentence delimiter (comma, period, etc.). For each character a copy of the Feature combination for an appropriate Segment, as determined from a user supplied mapping table, is transferred to a Work Buffer. All transformations within the Rule Part apply to the contents of this buffer, i.e. to sets of Feature combinations.

A context sensitive grammar describes the transformations to be carried out. Logically, the Rule Part may be divided into two phases: A Categorial and a Parametric phase. During the Categorial phase only Features or sets of Features are transformed. Specific Features may be changed, deletions are carried out by removing the relevant Feature matrix from the Work Buffer, and insertions are accomplished by copying complete Feature matrices from the definition tables to the Work Buffer.

When the program enters the second, parametric, phase all Feature matrices in the Work Buffer are mapped back onto the definition tables and the relevant acoustic Chunks are copied from the definition tables to the Work Buffer. Thus, changing Feature combinations within the Categorial phase has direct consequences for the choice of physical descriptions, i.e. the choice of allophones.

During the Parametric phase the contents of Variables and Parameter structures may be modified to reflect details of coarticulation, and adjusted values of duration and pitch may be computed to simulate the appropriate speech rhythm and intonation.

When all rules have been applied interpolations between the Parameter Target values of the Work Buffer are performed. These values are the final output of the conversion program.

Throughout the rule part special Print commands may be inserted. These commands will cause the contents of the Work Buffer at that stage to be printed in phonetic script, using the Character Names of the Segment definition tables (always assuming that the Feature combinations of the Work Buffer can be mapped back onto the Feature matrices of the definition tables). Alternatively, a special Trace facility may be turned on. This will cause the contents of the Work Buffer to be printed every time the Buffer is modified by a rule, thus giving a trace of the effect of individual rules.

3. SPEECH SYNTHESIZER

The Rule Compiler is not designed to operate with any particular speech synthesizer, although we have all the way been thinking in terms of a formant coded synthesizer. Instead, the Rule Compiler derives its knowledge about the synthesizer which will eventually use its output from a set of definition tables. This strategy will allow us to configure the same phonetic/ phonological rules for a variety of different speech synthesizers.

Currently no hardware synthesizer is connected to the computer, and we rely on the fast PDP-11/60 to simulate the whole process. Using a simulated synthesizer will, of course, give great flexibility in the choice of "machine", but the response time is relatively slow, and the system will have to be augmented with various hardware devices.

C. APPLICATIONS OF THE SPEECH SYNTHESIS SYSTEM

The speech synthesis system described here is planned primarily as a research system. Therefore the programs have been designed to be easily modified rather than being optimized for space requirements or speed of execution. However, we believe that the output of the Rule Compiler will be such that it should be possible eventually to transfer it to, for instance, a microprocessor ROM in order to build a sort of production version.

As it is, the system will be immediately available as a tool for testing phonological and phonetic hypotheses and for refining the exact description of Danish pronunciation. This kind of basic research is, of course, the background for any progress in the development of practical applications of speech synthesis.

Actually, for much of the phonological work to be done, speech output will not be required or even wanted. In these cases a phonetic transcription might well be the preferred output. Incidentally the phonological rule system could also produce the raw version of various kinds of pronouncing dictionaries for Danish.

REFERENCES

Basbøll, H. and Kristensen, K. 1974: "Preliminary work on computer testing of a generative phonology of Danish", *Ann. Rep. Inst. Phon. Univ. Cph. 8,* p. 216-226

Basbøll, H. and Kristensen, K. 1975: "Further work on computer testing of a generative phonology of Danish", *Ann. Rep. Inst. Phon. Univ. Cph. 9,* p. 265-292

Carlson, R. and Granström, B. 1974: "A phonetically oriented programming language for rule description of speech", *Preprints of the SCL-1974, 2,* p. 245-253

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

Holmboe, H. 1978: *Dansk retrograd ordbog* (Akademisk forlag, Copenhagen)

Holtse, P. 1973: "Identification and discrimination of closely spaced synthetic vowels", *Ann. Rep. Inst. Phon. Univ. Cph. 7,* p. 235-264

Holtse, P. 1974: "Preliminary experiments with synthesis by rule of standard Danish", *Ann. Rep. Inst. Phon. Univ. Cph. 8,* p. 239-251

Reinholt Petersen, N. 1974: "The influence of tongue height on the perception of vowel duration in Danish", *Ann. Rep. Inst. Phon. Univ. Cph. 8,* p. 1-10

Reinholt Petersen, N. 1976: "Identification and discrimination of vowel duration", *Ann. Rep. Inst. Phon. Univ. Cph. 10,* p. 57-84

Rischel, J. 1966: "Instrumentation for vowel synthesis", *Ann. Rep. Inst. Phon. Univ. Cph. 1,* p. 15-21

Rischel, J. 1968: "Constructional work on a function generator for speech synthesis", *Ann. Rep. Inst. Phon. Univ. Cph. 3,* p. 17-32

Rischel, J. and Lystlund, S.-E. 1967: "Speech synthesizer", *Ann. Rep. Inst. Phon. Univ. Cph. 2,* p. 34

Rischel, J. and Lystlund, S.-E. 1972: "A formant coded speech synthesizer", *Ann. Rep. Inst. Phon. Univ. Cph. 6,* p. IX-XXIX

Rosenvold, E. 1981: "The role of intrinsic Fo and duration in the perception of stress", *Ann. Rep. Inst. Phon. Univ. Cph. 15,* p. 147-166