

Text-To-Speech Conversion Technology


Michael H. O'Malley
Berkeley Speech Technologies

Text-to-speech (TTS) conversion transforms linguistic information stored as data or text into speech. It is widely used in audio reading devices for blind people. In the last few years, however, use of text-to-speech conversion technology has grown far beyond the disabled community to become a major adjunct to the rapidly growing use of digital voice storage for voice mail and voice response systems, which provide telephone information access. For example, text-to-speech technology can convert electronic mail to voice mail for audio access by phone. It can also permit field personnel to access large text databases, like parts inventories, by telephone.

The rapid expansion of text-to-speech technology comes in part from the advances in delivery methods and speech quality over the past 30 years. This article discusses the historical and theoretical bases of contemporary high-performance TTS systems and their current design. Because of space limitations, I have drawn examples mainly from Berkeley Speech Technologies' proprietary text-to-speech system, T-T-S.

What is TTS?

Any text-to-speech system consists of two major elements. Starting with the output, we need some type of sound-generating mechanism whose function is analogous to that of the human vocal tract. A mouth by itself cannot talk, so we also need



**Reading an English
text out loud is
currently the most
successful simulation
by a computer of a
complex human
mental function.
This article shows
how it can be done.**

a module whose input is the text or other linguistic information to be spoken and whose output drives the sound-generating mechanism. In modern technology, both of these components are software. We can implement them in such a way that they can run on many kinds of hardware platforms.

A schematic of the human vocal tract appears in Figure 1. Air pressure developed in the lungs flows through the trachea and the vocal folds in the larynx. This opening between the vocal folds is called the glottis, and the air flow as a function of time is called the glottal waveform.

For voiced sounds, the correct combina-

tion of air pressure and muscle tension causes the vocal folds to vibrate, generating a series of pulses of air. For aspirate sounds such as /h/, the vocal folds adjust to generate turbulent air flow. For many sounds, we also generate noise in a higher part of the vocal tract.

Some of the noise in fricative sounds (/s/, /z/, /f/, etc.) actually results from turbulent air flowing through a constriction between the tongue and the roof of the mouth. On the other hand, the main source of noise in plosive sounds (/b/, /d/, /k/, etc.) is a burst from the sudden release of air pressure built up behind a closure of the vocal tract, followed by a short period of friction. In English and certain other languages, "voiceless" stops such as /k/ are also followed by a period of aspiration noise when they occur in particular linguistic environments.

The noise generated at the glottis or elsewhere in the vocal tract is modified as it passes through the oral and nasal cavities and radiates from the head as a speech waveform. As it progresses along the vocal tract, some of the flow reflects backwards. In fact, some energy even reflects back through the glottis toward the lungs. These reflections induce resonances (frequencies at which the sound is reinforced) and antiresonances (frequencies at which the sound is absorbed).

The frequency of the glottal pulses is one of the important parameters characterizing voiced sounds, corresponding to the fundamental frequency or pitch of the voice. However, finer details of the vocal

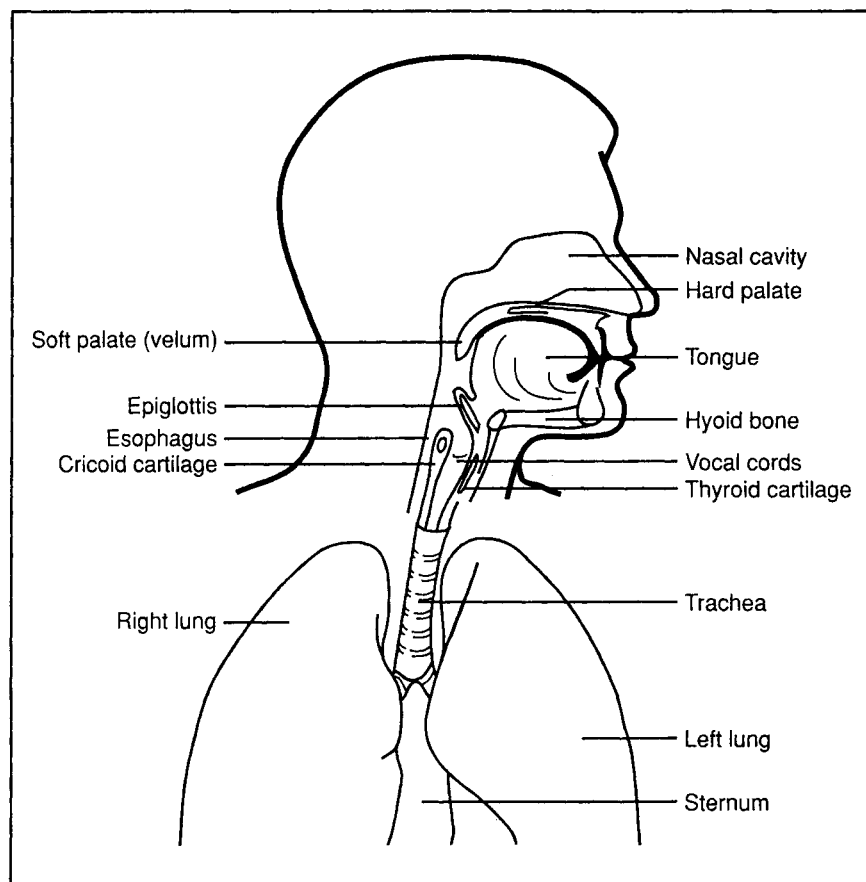


Figure 1. Schematic diagram of the human vocal mechanism.

fold vibrations as well as other elements of the sublaryngeal system also affect the overall sound quality.

Above the larynx, the shape of the tongue and lips and the size of the velar opening to the nasal cavity are major factors in determining the resonances of the vocal tract and thus the sound of the radiated speech waveform. Here again, even an obscure factor, such as the compressibility of the cheeks, can have some effect on the perceived sound.

The major features of the vocal tract and how they influence our perception of speech have been understood, at least in a general way, for years.^{1,2} Because of the tremendous complexity of the vocal tract, however, the attempt to understand and model the human speech production mechanism remains an active research topic in current acoustic and physiological phonetics research.³

Our current understanding does permit the synthesis of intelligible speech, but our models, especially in their dynamic behavior, are not yet adequate to make synthetic speech that is indistinguishable from human speech.

Vocal tract models based on the physical shape and the physiology of the sound production mechanism provide many scientific insights but are rather difficult to use. The complex relationship between these physical parameters and the resulting speech waveform involves nonlinear equations. A small change in one of the model's parameters often makes a major change in the resulting sound.

Speech can also be modeled in strictly acoustic terms as a continuously changing spectrum (see Figure 2). The most common parameterization in this domain is in terms of the resonances and antiresonances (poles and zeros), familiar from traditional engineering models of linear systems. In this model, the speech production process consists of an excitation source, presumably representing the glottal waveform, which drives a "filter," presumably representing the oral and nasal cavities.

The most important acoustic parameters for speech synthesis are the fundamental frequency of the glottal waveform and the frequencies of the first three narrow bandwidth resonances, or formants. For a typical male voice, the fundamental varies over

about an octave centered around 120 Hz, while the first three formants vary around 500 Hz, 1,500 Hz, and 2,500 Hz, respectively. The fundamental frequency for a female voice typically falls closer to 200 Hz, while the formant frequencies, which are inversely proportional to head size, measure about 10 percent higher.

This source/filter model of speech production is the most widely used model for acoustic phonetic research as well as for the most sophisticated applications of speech technology. However, it is only one of many possible parameterizations of the speech production mechanism.

For example, another model with wide applicability approximates the vocal tract by a cascade of short cylinders, each of a different diameter. This model is mathematically related to the widely used linear predictive coding (LPC) technique used in speech transmission systems and for recording human speech in compressed form on chips.

This multitube model has the advantage that it is easy to fit human speech to the model automatically, so we only need to transmit the model parameters rather than the full speech waveform. However, it is very difficult to relate the parameters of this multitube model to the usual scientific descriptions of speech. Therefore, this model has had limited applicability in the most sophisticated speech synthesis and recognition systems.

In my work at the University of Michigan in the 1960s, the real-time parametric voice module was actually a hardware analog filter. The hardware consisted of three voltage-controlled filters, a noise generator, and a pulse generator connected to an IBM 1800 computer with a total of only 8 kilowords of memory.

The filters modeled the resonances of the vocal tract and the pulse generator modeled the glottal waveform. We entered speech parameters by hand into the computer, which then sent them in real time to the hardware. By varying the parameters, we could study the effect of changing the voice model. The system helped in studying human speech production and perception, but was not sufficiently powerful to support the automatic conversion of text into speech in real time.

One major factor in the progress toward this goal has been the development of digital simulation of the kind of hardware used in the early systems. In general, an analog resonator can be simulated digitally by a very simple program consisting of addition, multiplication, and some storage

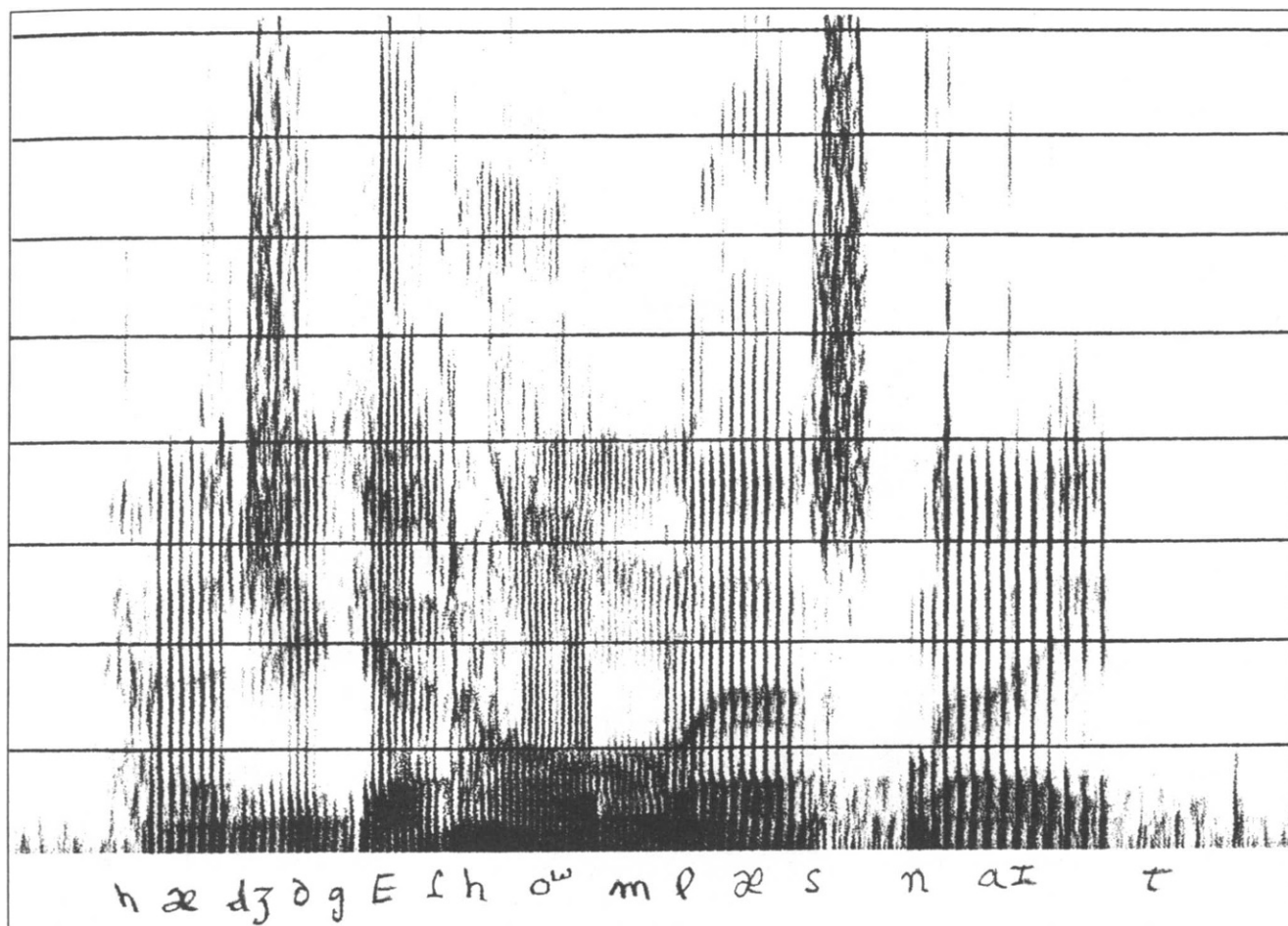


Figure 2. Speech modeled in acoustic terms as a continuous spectrum: "How'd you get home last night?"

locations. Approximately 20 multiplications are required in order to generate a single speech sample for a typical vocal tract model.

"Telephone quality" speech requires 8,000 samples per second, although most speech researchers prefer a somewhat higher bandwidth — 10,000 to 20,000 samples per second. With the computing power available by the 1970s, researchers could replace analog human sound production models by digital synthesizers in almost all research applications. Even when they lacked the computing power necessary for real-time synthesis, developing a more powerful, all-digital model and running it in batch mode was preferable to trying to make do with an inadequate hardware analog of the vocal tract. Thus, for more than 10 years most speech synthesizers have been software.

In 1978 Texas Instruments developed the "Speak & Spell" LPC chip. This chip can perform an integer multiply/add in less than 5 microseconds and therefore can

implement a 10-pole lattice filter digitally. Unfortunately, this chip implements the multitube vocal tract model, with the consequent difficulty in generating synthetic parameters. However, because we can easily extract these filter coefficients from natural human speech, these chips found widespread use delivering speech in a number of products such as toys and talking cars. Of course, human voice stored and compressed by algorithm is not truly "synthetic" in the ordinary sense of "artificial" or "not of natural origin," but is more properly considered a low-bit-rate recording. True synthesis implies a model of a speaker, not an actual speaker.

During the 1980s, easily programmable digital signal processing chips became available. These chips, such as the Texas Instruments TMS320, could perform a multiply/add cycle in less than one microsecond. This meant that, at least for audio frequencies, even the most complex research voice models could be programmed to run in real time in a relatively inexpen-

sive system. After this, practical hardware or computational limitations on the implementation of a synthesis voice model no longer existed.

However, the voice model constitutes a very small part of the text-to-speech synthesis problem. A voice model requires from a few hundred to at most a few thousand computer instructions. In contrast, the text-to-parameter model for a high-quality synthesizer requires from 100 to 1,000 times as much memory as the voice model to capture the extensive linguistic knowledge required for true synthesis.

It is interesting to compare the relative computational complexity in current speech recognition and speech synthesis systems. A speech recognition system must have some kind of speech analysis module, which functions as an "inverse" of the voice model of a speech synthesis system. However, this analysis task is more difficult than the corresponding synthesis because the signal is "real" rather than the result of a simplifying model. The better

speech recognition models are related to the synthesis voice model, but they normally occupy a larger fraction of the available computer resources.

A speech recognition system must also have a "language model" to guide the mapping from the analysis parameters to the recognized text. In some sense, this module is the inverse of the text-to-parameter conversion module in a speech synthesis system.

In actuality, the language module for a speech recognition system is often rather simple and "algorithmic" rather than knowledge intensive. In fact, some recognition language models contain no linguistic information at all. They work with any language or even with nonlanguage sounds. However, even these simple language modules tax the power of current computer systems.

Generally speaking, text-to-speech systems are limited by our current knowledge of linguistics. Speech recognition systems are more limited by computing resources and by our ability to apply the linguistic knowledge available.

The text-to-parameter conversion module takes as input an English text (or, more usually, structured information from a database) and generates the parameters that can then drive a vocal tract model. We can think of this module as a model of how we convert linguistic information into parameters that drive the speech production mechanism. In other words, how do we read aloud from a printed text?

Of course, we know much less about how we process language than we know about how the human vocal tract works. However, we know a great deal about the structure of language. Linguistics is an old science and, especially over the past 30 years, researchers have developed a number of models for various parts of language. The text-to-parameter module might not really provide a good model of how we generate speech, but it does incorporate a great deal of knowledge about the English language.

The process of converting text into speech parameters breaks down into a number of stages, as shown in Figure 3. The following sections summarize these various subprocesses.

Text normalization

Actual texts have a great deal of symbolic material such as numbers, abbreviations, acronyms, and information signaled

by graphic layout. The first step in a text-to-speech conversion system converts such information into a standard text format. The complex conversion process involves various types of local parsing. While the details lie beyond the scope of this article, some examples can illustrate the various difficulties.

First, consider the different ways we can read numbers. For example, we can read the sequence "415" in three different ways depending on whether it is an area code, part of an address, or a dollar amount. Furthermore, the rules for converting numbers into spoken English will differ in the United States and in England.

The pronunciation of various abbreviations is often determined by context. For example, the pronunciation of letter/punctuation mark sequences differs markedly depending upon where they occur:

- Dr. Jones lives on Jones Dr.
- St. James St.
- Jan. 22 is my wife's birthday; her name is Jan.

The effect of punctuation marks is also often context determined. For example, a period at the end of a sentence has a major effect on sentence prosody. But a period at the end of an abbreviation, in a decimal number, or after a middle initial in a name has a very different significance and must not be misinterpreted as a sentence termination in pronunciation. Columnar lists of items, pronounced as if they ended with a period, often do not have any punctuation at all. In this case, the text-to-speech system must recognize the two-dimensional shape of the text to pronounce it correctly.

All of the above phenomena can be handled reasonably successfully, but they require an extensive, nonalgorithmic computer program. Such a program captures facts about normal American English written text that almost every literate person knows implicitly. It probably does not provide a good model of our mental processing, but it does model human knowledge in the same sense that a good chess program models a chess master's knowledge of chess.

Word pronunciation

How we pronounce English words today depends on 2,000 years of history, dozens of wars, and many population migrations. Because of this, English has by far the most

complex relationship between spelling and word pronunciation of any alphabetic language.

We all learned the rule that a final "e" makes the preceding vowel tense — the rule that turns "dud" into "dude." However, a reasonably accurate text-to-speech system requires several thousand such rules to handle English.

Any language with an alphabetic writing system has rules that convert spelling into sound. However, words borrowed from another language tend to reflect the rules of that language, modified to fit the patterns of the new language. The complexity of English spelling derives from its diversity of sources.

For example, the large number of English words borrowed from Romance languages tend to be stressed on the third syllable from the end ("intelligent"). However, some affixes move the stress location ("intelligibility"), while others do not ("intelligently"). English speakers often stress names and other words of recent non-English origin, such as "Dukakis," on the second syllable from the end, regardless of their stress in their original language.

You might think that pronunciation rules could be replaced with a large dictionary: just put the exact pronunciation of each English word in a table and then look it up. In fact, a text-to-speech system must use a dictionary to pronounce certain exceptional words. However, the ordinary English speaker encounters a large number of words and an even larger number of names. For example, it has been estimated that the average American high school student might encounter any one of 500,000 different words.⁴ In the United States, the 1970 census listed more than 2 million different last names, any one of which might occur in a typical computer database. Furthermore, our culture adds new words and new names every day, which means that a dictionary would always be out of date. A large dictionary, by itself, cannot provide a solution.

The word pronunciation module for a high-performance text-to-speech system must combine a dictionary with sophisticated pronunciation rules. The dictionary handles words with a syntactic or context-dependent effect, as well as words not regular enough to justify a rule. The rules then handle the rest of the words, including new words and names. Of the several million words and names possible, the majority will thus be pronounced correctly by an efficient system.

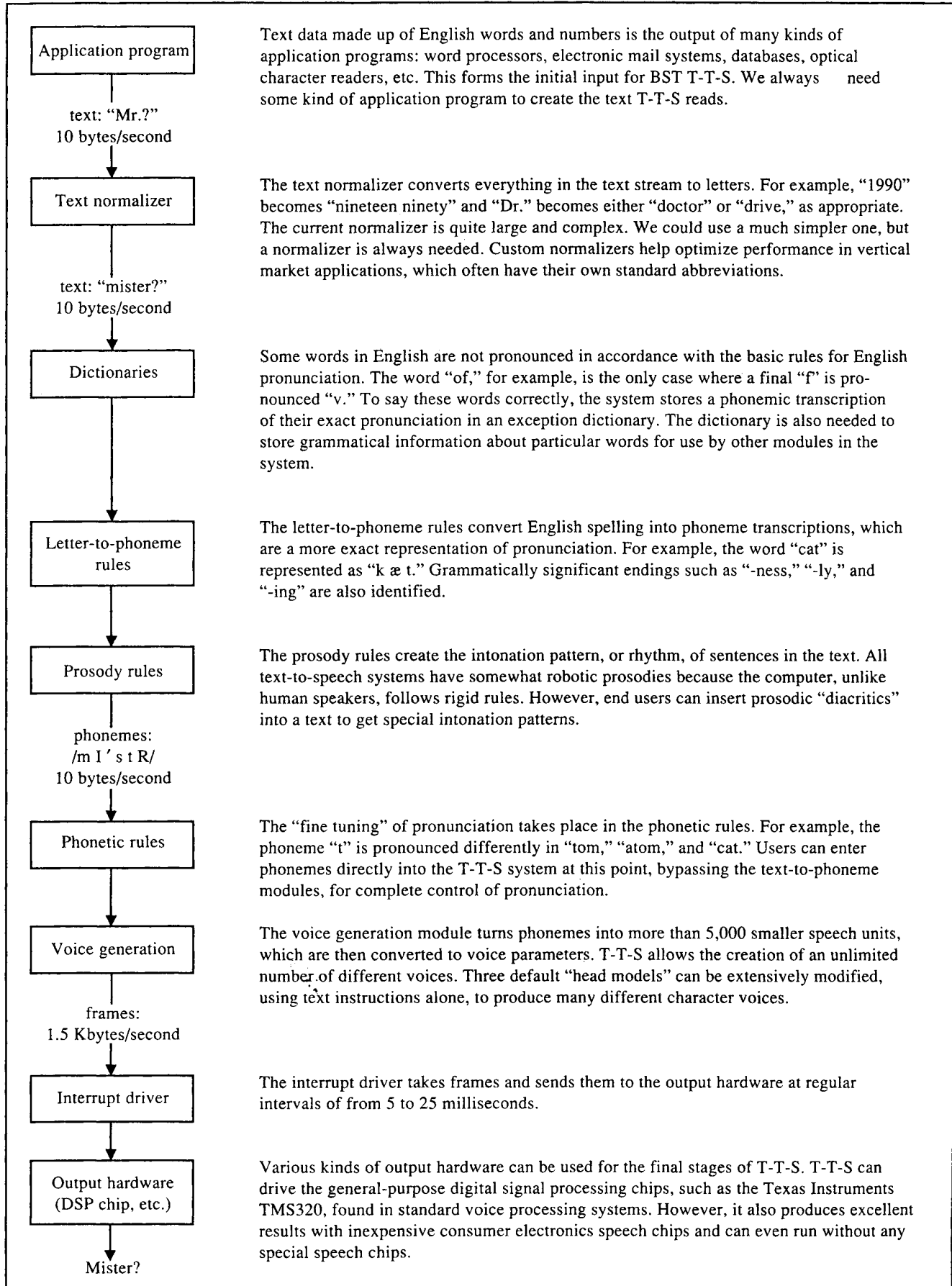


Figure 3. The process of converting text into speech parameters in Berkeley Speech Technologies T-T-S system.

Prosodies

The prosodic component of speech involves such phenomena as rhythm, intonation, and the emphasis or de-emphasis of particular words. For any given text, a human speaker might produce thousands of possible prosodic interpretations, but an even larger number would not be considered natural under reasonable circumstances.

As you listen to speech, certain words will seem more prominent than others. Often, these words carry the information focus of the message. On the other hand, such word stress patterns might simply signal that the word is part of a multiword compound: "Jones Street"; "history teacher."

Speech usually divides into a number of shorter phrases. Some of the division points will be accompanied by actual silence, while some will be marked only by an apparent slowing. However, each phrase will contain at least one of the emphasized words.

In addition to emphasized words and division into phrases, speech has an intonation contour. For example, some of the phrases typically end either with a falling pitch or with a fall followed by a low rise — a "period" or a "comma" intonation.

If part of a conversation, speech will exhibit quite elaborate prosodic phenomena that might carry a significant portion of the message's meaning. But speech as read from a text, especially by a person neither interested in nor knowledgeable about the text, will provide much more regular prosodic information.

Prosodies have been studied extensively in linguistics. However, they seem to be one of the most complex parts of language, and the information needed to generate them correctly from a text is not always available from current computational linguistic technology.

For example, prosodies depend in part upon syntactic structure. Usually, a prosodic break occurs between a long subject and the predicate in an English sentence. Because no current natural language parser is robust enough for use with general written English texts, we cannot always find this break reliably.

Prosodies generally prove the most difficult part of language for a text-to-speech conversion system. A very good prosodic component would probably require that the computer actually "understand" the text. However, an adequate if rather monotonic or dull reading is possible using the

most neutral or general case of current prosodic models. For example, textual punctuation marks usually indicate prosodic boundaries, and the last word in a phrase normally gets the primary emphasis. Articles and prepositions are de-emphasized and associated with minor phrase boundaries.

The resulting speech sounds somewhat boring, but most attempts to make it more lively result in some sentences having a foolish-sounding prosody. Most people might also, when presented with a text and asked to read it aloud, use something closer to the prosody produced by a text-to-speech system than they would use in speaking naturally.

Phonetic rules

After we have applied the text normalizer, word pronunciation rules, and prosody rules, the utterance consists of a string of phonemes. Phonemes are the symbols used by some dictionaries to represent pronunciation. Although a long way from actual speech sounds, they represent the finest distinctions that most people not trained in phonetics normally notice. The concept of "the phoneme" covers a range of recognizable expressions, not a point. The distribution pattern for various phoneme realizations is part of what defines dialects. Ordinary, casual speech represents phonemes in about the same way that messy human handwriting represents the abstract ideal graphic letters of the alphabet.

The phonetic rule module takes phonemes as input to produce a detailed description of the sounds of the utterance. The rules in this module do things such as assign a duration to each phoneme according to context, for example, lengthening the final vowel in a phrase by up to 50 percent. Other rules remove parts of phonemes, for example, deleting the aspiration of a stop such as /k/ in a number of different contexts.

Phonetic rules play the major role of describing the co-articulation between phonemes. Each phoneme strongly influences the parameters in the adjacent phoneme. In some cases, this influence might extend over several adjacent phonemes.

A great many phonetic rules must be programmed in a text-to-speech conversion system. Without these rules, the speech output might be somewhat recognizable, but it would sound very unnatural. In American English, for example, the

phoneme /t/ should sound quite different in the words "Tom," "cat," and "butter."

In rapid human speech, the application of phonetic rules is sometimes quite extreme. Consider, for example, the classic case of the phrase "Did you eat yet?" This phrase can reduce to something closer to "Chee chet?"

People can make such reductions because they understand and get feedback from their communication environment. Thus, they can judge the degree to which reduction is appropriate. If it is not appropriate, they can always speak more formally.

While we could certainly program such "fast speech" rules for a text-to-speech conversion system, it is probably more appropriate to maintain a more formal style of speech. For example, "want to" or "thank you" often reduce to "wanna" or "thank ya" when spoken. However, an electronic mail system that read its messages aloud over the telephone in such an informal style would seem odd and out of place in a business context.

Voice tables

The phonetic rules provide a detailed phonetic description for an utterance. The voice table module converts this description into numeric targets for use by the voice model. The phonetic rules in conjunction with the voice tables are the primary determinants of voice intelligibility.

Intelligibility is the likelihood that a human listener will be able to identify a particular word spoken. In one standard test,⁵ a modern, high-quality text-to-speech system typically scores approximately 95 percent. In contrast, low-end systems score from 60 to 75 percent. Careful human speakers typically produce about half the listener error rate of the best text-to-speech systems, although some communication channels carrying human speech elicit scores as low as 85 percent on this test.

One major function of the voice tables is to handle differences in bandwidth. Speech researchers have usually assumed a minimum speech bandwidth of 5 or 6 kHz. Unfortunately, telephone system designers have traditionally employed a bandwidth of 3.5 kHz, and even the newest digital telephones have perpetuated that bandwidth.

Important features of fricatives and stops occur in the frequency range of 4 to 6 kHz. Human speech is so redundant that

generally we don't miss this frequency range. However, text-to-speech systems can easily suffer losses unless we account for this difference in frequency range.

Our solution is to have different sets of voice tables for different frequency ranges. For applications involving telephones, some of the frequency components above 4 kHz can be mapped to a lower frequency, potentially contributing to overall speech quality and intelligibility.

Hardware implementation

In the work that we at Berkeley Speech Technologies have done on our own text-to-speech system, a major goal has been to develop a highly portable version of text-to-speech conversion software. We want to be able to deliver the same high quality of speech on any hardware platform that meets certain minimum requirements.

In terms of computational complexity, our text-to-parameter conversion process requires from 250 to 350 kilobytes of memory and a processing power of about 0.2 million instructions per second. Since this module is not very computationally demanding, we coded it entirely in the C language.

The voice model, on the other hand, requires less than 1 Kbyte of code but on the order of 1 MIPS. Since this module is small and usually time critical, we almost always encode it in assembly language.

The design of our software, coupled with recent advances in available hardware, has allowed us to successfully implement text-to-speech capability on a wide variety of hardware platforms. These include

- A pocket-sized talking dictionary with an 83,000-word vocabulary. It contains no digital signal processing chip, but synthesizes words in real time in an 8-MHz NEC V20 microprocessor.
- A telephone voice-response board that simultaneously generates 16 channels of speech from text in real time, then converts the speech waveforms into 32 kilobit-per-second ADPCM (adaptive delta pulse code modulation) coded speech.
- A stand-alone text-to-speech conversion unit with a built-in speaker and a 12-volt power supply, designed for speaking text messages sent to trucks by satellite.
- A personal speech-output device for blind people. Using CMOS hardware makes possible portability and battery-powered use with laptops. It speaks up to 700 words a minute.

Other approaches

The principal high-end commercial text-to-speech systems for American English are Digital Equipment's DECtalk, Centigram's Speech Plus, and our own BST T-T-S. Both DECtalk and Speech Plus are based on the work of Dennis Klatt at the Massachusetts Institute of Technology.⁶

Broadly speaking, Klatt's approach resembles the approach described in this article, based on an elaborate, fully synthetic, formant-based model of voice production. While the systems differ in many ways, the overall effect is roughly similar.

At the other end of the spectrum, a number of "toy" quality synthesizers have appeared on the market for use with low-end personal computers. Generally, they have relied either on recorded waveforms or on one of the "speech synthesis" chips.⁷

The main problem with this approach is that these chips are programmed to take phonemes as input and to generate between 64 and 128 presynthesized segments. However, these chips have not implemented the large number of phonetic rules that map phonemes into speech sounds, so have been fairly unintelligible.

Another approach, under the names "demi-syllable," "diphone," or "dyad" synthesis, involves recording several thousand segments of human speech using some form of LPC encoding.⁸ These segments represent all of the theoretically possible transitions between adjacent phonemes or sequences of phonemes. Such an approach eliminates the need for some of the co-articulation rules but in return makes some of the other phonetic rules more difficult to apply. Some good research implementations have used this approach but, I believe, it does not solve any significant problems. Moreover, it has not been adopted in the highest quality commercial text-to-speech systems.

Unrealistic expectations for dramatic future improvement in text-to-speech technology sometimes arise from an unsophisticated view of the complex linguistic information involved. Improvement will continue at the same slow, steady pace that has produced incremental progress in accuracy, intelligibility, and naturalness over the past three decades. However, we can expect faster progress in methods of delivering the technology. ■

References

1. G. Fant, *Acoustic Theory of Speech Production*, Mouton and Co., 's-Gravenhage, Netherlands, 1960.
2. J.L. Flanagan, *Speech Analysis Synthesis and Perception*, 2nd ed., Springer-Verlag, New York, 1972.
3. D.H. Klatt and L.C. Klatt, "Analysis, Synthesis, and Perception of Voice Quality Variations Among Female and Male Talkers," *J. Acoustical Society of America*, Vol. 87, Feb. 1990, pp. 820-857.
4. J.B. Carroll et al., *Word Frequency Book*, American Heritage, New York, 1971.
5. J.S. Logen, B. G. Greene, and D.B. Pisoni, "Segmental Intelligibility of Synthetic Speech Produced by Rule," *J. Acoustical Society of America*, Vol. 86, No. 2, Aug. 1989, pp. 566-581.
6. D.H. Klatt, "Review of Text-to-Speech Conversion for English," *J. Acoustical Society of America*, Vol. 82, No. 3, Sept. 1987, pp. 737-793.
7. I.H. Witten, *Principles of Computer Speech*, Academic Press, New York, 1982.
8. J.P. Olive, "Rule Synthesis of Speech from Diadic Units," *Proc. 1977 IEEE Int'l Conf. Acoustics, Speech, and Signal Processing*, Vol. ICASSP-77, pp. 568-570.



Michael H. O'Malley founded Berkeley Speech Technologies (initially called Berkeley Systems Works) in 1979. As president and chief scientist at BST, he has been engaged in the development of advanced text-to-speech systems since 1980. He has engaged in speech research since 1961, when he worked for IBM Research on a text-to-speech project while still a student at the California Institute of Technology. He did his PhD work in the University of Michigan Program in Communications Sciences, where he studied computer science, electrical engineering, linguistics, and biological systems.

From 1968 through 1973 O'Malley directed the Phonetics Laboratory at the University of Michigan as a member of the faculty. He was a principal investigator in the ARPA Speech Recognition Project, which he began at Michigan and continued at the University of California at Berkeley as a member of the UC Computer Science Department.

Readers can contact the author at Berkeley Speech Technologies, 2409 Telegraph Ave., Berkeley, CA 94705.