

太田 治徳  
H. Ohta

高橋 勉  
T. Takahashi

原 義幸  
Y. Hara

文音声合成 (Text-To-Speech: TTS) は、漢字かな混じり文から音声を生成する技術である。TTS を利用したヒューマンインタフェースを提供するソフトウェア環境を、Microsoft (注1)社の最新のオペレーティングシステム (OS) である日本語 Windows (注2) 95 上で構築した。

TTS 処理部をすべてソフトウェアで実現し、Microsoft 社が提案している API (Application Programming Interface) のサブセットを備えている。その上位に、支援ツール群を提供することにより、電子メール、ドキュメント、時刻などの読上げ、文章校正支援、イベントの通知などを可能にした。パソコン (PC) では、各種のアプリケーションから TTS を利用できなければならない。そのため、標準 API の提供が必須(す)であり、さらに各種設定や辞書登録の容易さ、高品質な音声の出力が要求される。

Text-to-speech (TTS) is a technology that synthesizes speech from Japanese text. This paper describes our own TTS software environment that has been designed for Microsoft's latest Windows95 operating system (OS).

The TTS procedure is accomplished in software, and we have provided our own subset to the Microsoft speech application programming interface (API). By stacking various tools on the TTS engine, we have made possible e-mail reading, document and spreadsheet proofreading, a speaking clock, and notifications to the user of asynchronous events. Applications on a personal computer should be able to easily take advantage of the capabilities of TTS. Thus, it is imperative to provide a standard API set, an easy-to-use dictionary, configuration utilities, and high-fidelity voice output.

## 1 まえがき

音声は、人間にとってもっとも自然な情報伝達手段である。このため、コンピュータと人間とのインタフェース (ヒューマンインタフェース) に音声を利用する研究は古くから行われてきた。TTS は、文字列から音声を生成する技術である。

当社は、これまで TTS を応用した種々のシステムを開発してきた。しかし、これらは音声を処理するために特殊なハードウェアを必要とし、限られた分野でしか使用できなかった。

近年、PC のヒューマンインタフェースとして音声が目まされている。背景には、OS にオーディオ信号を扱えるマルチメディア機能が追加されたことや、サウンドボード、スピーカを備えたマルチメディア PC の普及がある。特に、PC に高速の CPU が使用され、音声処理がソフトウェアだけで安価に実現できるようになってきたことが大きい<sup>(1)</sup>。

ここでは、TTS の基本技術を紹介した後、PC への適応のために TTS が備えなければならない機能を分析する。さらに、

(注1) Microsoft は、Microsoft 社の商標。

(注2) Windows は、Microsoft 社の商標。

PC 上での実現例として、TTS を利用したヒューマンインタフェースを提供するソフトウェア環境を、日本語 Windows95 上で構築したので、その概要を説明する。

## 2 TTS 技術

TTS 処理は、図1に示すように、大きく言語処理部と音声合成部の二つに分けられる。言語処理部は漢字かな混じり文から音韻系列と韻律情報を生成する。音声合成部は、これらを基に音韻パラメータと韻律パラメータを生成し、合成器を駆動して合成音声を出力する。

以下では、処理の詳細を説明する。

### 2.1 形態素解析

形態素解析は、約 11 万語の単語辞書と接続情報を用いて、入力文章を単語単位に分割する。すなわち、入力文章と単語辞書とを照合して得られるすべての単語系列候補を求め (総当たり法)、その中から、接続規則を参照して文法的に前後に接続できる組合せを出力する。単語照合に総当たり法を採用することによって高精度の解析を実現している。単語辞書には、

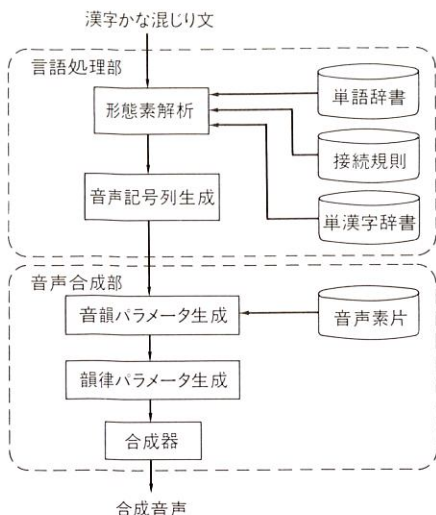


図1. TTS 処理の概要 漢字かな混じり文を音声に変換する処理の流れを示す。

TTS procedure

見出し語、読み、品詞、アクセント型などが登録されており、分割された単語はひらがなの読みに変換される。

一方、単語辞書に存在しない語句（未知語）は、その語句の前後の情報と単漢字辞書の見出し語および頻度情報を利用して漢字の読みを決定する。

2.2 音声記号列生成

ここでは、単語単位に分割された語句に基づいて、音韻系列とアクセント型などの韻律情報からなる音声記号列を生成する。ここで行う主な処理を次に挙げる。

- (1) 読みの変形 長音化 “一”, “へ” → “え”, “は” → “わ”などを処理する。
- (2) 単語の結合 分割された単語を一つのアクセント句になるように結合し、さらに品詞とその接続規則を用いてアクセント句のアクセント型を決定する。
- (3) ポーズ挿入 ポーズ挿入は、音節の数と品詞から決定される。また、ポーズの長さは文節間の接続規則を用いて決定される。
- (4) 鼻濁音化, 無声化 ガ行の鼻濁音化および母音の無声化は、音韻環境から決定される規則を用いて対応する。

図2は、上述の言語処理の一例を示している。

2.3 音韻パラメータ生成

合成音声の音質、明りょう度および滑らかさは、音声合成に用いる分析手法、合成の単位およびその接続規則によって変わる。当社は、単音節 (CV) を単位とするケプストラム方式を採用している。この分析手法は、音声スペクトルの極、ゼロを同等に扱い、各音声素片間を接続するときの補間特性がよいため、CV を合成単位とする場合にも明りょうで滑らかな音声を生成する。

ここでは、最初、音声記号列中の音韻系列に対応するパラメータを音声素片から取り出すとともに、音節時間配置パラ

(原文)  
今日は天気がよくて公園へ行って本を読みます。

(形態素解析)  
キョウ/ハ/テンキ/ガ/ヨイ/ノ/デ/コウエン/ヘ/  
イツ/テ/ホン/ヲ/ヨミ/マス

(音成記号列生成)  
キョウ/ワ./テンキ/ガ/>/ヨイ/ノ/デ./  
コウエン/エイ/ッテ./ホン/ヲ/ヨミ/マ`/ス//

“>” はアクセントの位置, “/” はアクセント句の区切り,  
“.” はポーズを示す。< > は鼻濁音, ( ) は無声化音。

図2. TTS の言語処理の例 単語辞書と各種規則によって得られる音声記号列から、自然な合成音声生成される。

Example of Japanese text processing with TTS

メータを基に音韻長を決定する。この音韻長は、アクセント句における素片の種類、素片の順序、素片の数に基づいて、素片ごとに子音 (C) および母音 (V) の継続時間の長さを決める。音声素片には、アナウンサの音声を改良ケプストラム法<sup>(2)</sup>によって CV 単位に分析したパラメータと、各素片固有の音節時間配置パラメータ、および素片のフレームごとに V/U 情報 (有声/無声情報) が格納されている。

次に、各素片間を補間して接続することにより、音韻パラメータを生成する。補間は、素片どうしを滑らかに接続するための処理で、11 種の補間結合規則に基づいて行われる。

2.4 韻律パラメータ生成

音声中の韻律は、話調の降下成分とアクセント成分の二つの成分に分離できる。話調の降下成分は、息継ぎが生じてから次の息継ぎが生ずるまでの基本周波数 (ピッチ) が徐々に低くなる成分であり、一方、アクセント成分はアクセント型によって決まる。アクセント型は大きく分けて、平板型 (0 型)、頭高型 (1 型)、n 型に分類される。これらのピッチパターンの軌跡の位置と高さは、あらかじめテーブルに格納されており、音韻系列とアクセント型との組合せによって修正される (アクセント成分)。このアクセント成分とイントネーション成分とを対数軸上で足し合わせることで、ピッチパターン (韻律パラメータ) が生成される。

2.5 合成器

図3に音声合成器の構成を示す。ここでは、まず韻律パラメータに従って音源を生成する。

音声の音源は、無声部をランダムノイズ、有声部をインパルス列で近似できる。そこで、無声部には M-系列 (ランダムデータ) の発生器を、有声部にはピッチ周期 (IP) を入力とするインパルス列発生器を用いている。

次に、音源の振幅は図3中に示す式で与えられる。式は音声パワー  $C_0$  (ケプストラムパラメータの 0 次) とピッチ周期  $IP$  の値によって振幅が変化することを表しており、声の高さが高くなるほどその大きさが小さくなる。合成部はケプストラムパラメータを直接その係数とする LMA (Log Magnitude

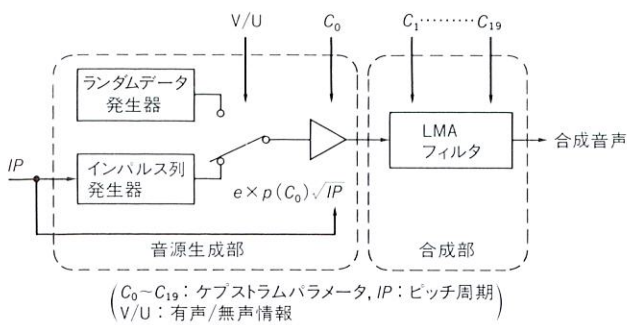


図3. 音声合成器の構成 LMA フィルタの採用により、良質の音声生成される。

Configuration of speech synthesizer

Approximation) フィルタを採用している<sup>(3)</sup>。この合成フィルタは少ない演算量で良質の音声を生産する。

### 3 PC への適応

ここではPCでの用途を分析し、そのために必要なTTS技術を考える。

#### 3.1 PCのヒューマンインタフェースとしての用途

TTSをPCのヒューマンインタフェースとして利用する場合の主な利点は次の3点である。

- (1) 表示装置を見ることができない状況での、PCからの反応を受けるただ一つの方法である。表示装置を見る必要がないので、他の仕事を同時に行うことができる。
- (2) 動的な文への適応に最適。つまりあらかじめ録音しておくことが現実的でない文や状況でその特長を發揮する。
- (3) 入力データが文字コードなので音声データに比べ数けたも少ない情報量で済み、省スペースになる。

その利点を生かした具体的な用途には次のものがある。

- (1) 電子メール、ドキュメント、データなどの読上げ
- (2) 時刻の読上げ
- (3) 校正支援 視覚での校正で見落とすタイプミスをチェックできる。表計算ソフトウェアなどへの適応。
- (4) イベントメッセージの通知 プログラム実行中のエラーメッセージ、メールの着信や印刷ジョブの完了などの読上げによる利用者への通知。

#### 3.2 TTSエンジンに求められる機能

3.1節で述べた用途をPCで実現するために、TTSエンジンとして備えなければならない点がある。TTSエンジンとは、漢字かな混じり文を合成音声に変換するPCにおけるモジュールのことをいう。

- (1) 標準APIの提供 特定のアプリケーションではなく、市販のアプリケーションから容易に使用できなくてはならない。そのためには、APIの標準化が重要であり、そのAPIを備えたTTSエンジンの開発が必要である。

- (2) 変換の正確さ 電子メールやさまざまなデータを読み上げる必要がある。文法的に明確でない文章の読上げや、各種の省略語、技術用語へ対応しなければならない。
- (3) 高音質 PCでは、録音された音声データが豊富であり、通常それらを再生している。TTSの生成する音声は、それらに著しく劣るものであってはならない。人間の声にきわめて近い韻律が要求される。
- (4) 設定の容易さ 各アプリケーションから容易に、読上げ音声の設定や読みかたの指定ができなければならない。読上げ音声の設定とは、声の種類(男女の選択)、発語速度、声の高さ、音量などを指定すること。読みかたの指定とは、数字、英単語、特殊記号などに対して行う。例えば、数字をけた読みするか、棒読みするかを指定する。
- (5) 辞書登録の容易さ 利用者が新しい語を読上げのために登録するとき、TTSの場合、発音やアクセントを指定する必要があり、その操作が容易でなければならない。

### 4 日本語 Windows95 上での TTS システムの実現

3章で述べた考察を基に、TTSを利用したヒューマンインタフェースを提供するソフトウェア環境を、日本語 Windows95 上で構築した。以下、その概要を説明する。

TTSシステムは、図4に示すように大きくユーザインタフェース部とTTSエンジン部の二つに分けられる。

ユーザインタフェース部は、TTSを標準にサポートしている市販のアプリケーションが現状少ないので、ユーザがTTSを利用するために必要な支援ツール群を提供している。

TTSエンジン部は、TTS処理を行うモジュールで、上位ソフトウェアに対してTTS機能を使用するためのAPIを提供する。APIは、Microsoft社が提案しているSpeech API<sup>(4)</sup>のサブセットになっている。

両者とも Win32<sup>(註3)</sup>に準拠したソフトウェアである。

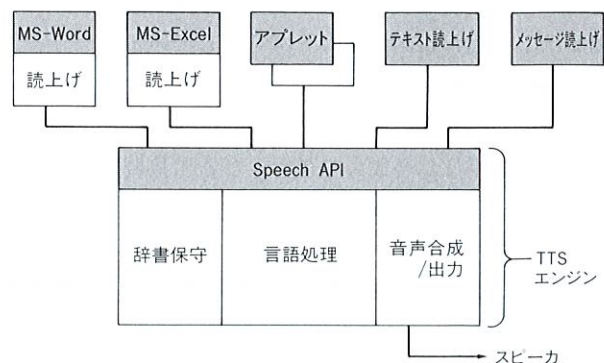


図4. 日本語 Windows95 上での TTS システム Speech API のサブセットを備えた TTS エンジンと支援ツール群を提供している。

TTS software environment in Windows95

#### 4.1 TTS エンジン部

TTS エンジン部は、API を提供する Speech API 部と、実際の TTS 処理を行う文音声合成エンジン部の 2 層から成る。

**4.1.1 Speech API 部** 文音声合成エンジン部の機能をユーザに提供するためのインタフェースモジュールである。ユーザはこの API を介してだけ文音声合成エンジンの機能を使用することができる。テキストあるいは音声記号列の音声への変換、音声出力の停止/再開、音声属性の各種設定、ユーザ辞書登録などの機能を提供する。

Microsoft Speech API のサブセットになっているこのインタフェースは、OLE2<sup>(5)</sup>に基づき定義されており、上位アプリケーションは OLE Component Object Model<sup>(5)</sup>を通してアクセスする。

実装は、エンジン DLL (Dynamic Link Library) の上層部にエンジンラッパーがあり、これによりエンジンは、OLE2 を通じてオブジェクトとして扱うことができる。今回は、エンジン自身を表すオブジェクトと、そのオブジェクトを操作するためのインタフェースを実現した。

オブジェクトとすべてのインタフェースは、すべて C++ 言語で書かれているが、OLE2 の機能によりどのような言語で書かれたプログラムからでも呼び出せる。

**4.1.2 文音声合成エンジン部** Speech API で提供する機能の大部分を実行するモジュールで、32 ビット DLL になっている。

#### 4.2 ユーザインタフェース部

日本語 Windows95 上でユーザが TTS を利用するために必要な次の支援ツール群を提供する。

- (1) テキスト読上げ アプリケーションプログラムであり、Speech API を使用して次の動作をする。  
テキストファイルの読上げ、テキストファイルの作成、クリップボードの読上げ、OLE サーバ機能 (読上げ文書の埋込みおよびリンクのサポート) の動作を行う。さらに、音声属性の各種設定、ユーザ辞書登録/削除/更新/検索を行う。
- (2) アプレット 入力したキーおよび計算結果を読み上げる音声電卓アプレット、時刻を読み上げる音声時計アプレット、キーボードから入力されたキーコードを読み上げるキーボード読上げアプレットを提供する。
- (3) MS-Excel 読上げ Microsoft Excel のデータを読み上げるためのマクロ機能を提供する。
- (4) MS-Word 読上げ Microsoft Word の文書を読み上げるためのマクロ機能を提供する。
- (5) メッセージ読上げ システムの状態を常時監視して、システムからのエラーメッセージを読み上げたり、システムの開始/終了時などに特定のメッセージを読み上げるプログラムである。

(注 3) Win32 は、Microsoft 社の商標。

## 5 今後の課題

今回、日本語 Windows95 上で TTS システムを実現した。しかし、まだ PC としての開発の緒についたばかりである。

今後は、各種の市販アプリケーションによる Speech API 利用が期待できる。そのときは、エンジンの品質がさらに問われ、自然な音声を生成するための規則の改良、日本語解析技術の向上による変換の正確さが要求される。

また、PC に音声モデムが装備され、電話ライン経由での電子メールやファックスの読出しのためのインタフェースとして音声を利用したアプリケーションが出てくるであろう。そのときは、文音声合成に加え、音声認識も必要とされる。

## 6 あとがき

長年にわたり研究所レベルで培われてきた要素技術を PC に適用した。de-facto で成り立つ PC の世界に当社固有の技術を使用することは困難と思われてきたが、インタフェース定義が明確であれば、機能の実現には固有の技術を遺憾なく発揮でき、これによる差異化が期待できる。今後とも、固有の要素技術を PC に積極的に取り入れる所存である。

## 文献

- (1) 桃崎浩平、他：パソコン用文一音声変換ソフトウェアの開発、日本音響学会講演論文集、pp.327-328 (1994)
- (2) 今井 聖、他：改良ケプストラム法によるスペクトル包絡の抽出、電子情報通信学会論文誌、J62-A、4、pp.217-223 (1979)
- (3) 今井 聖：対数振幅近似 (LMA) フィルタ、電子情報通信学会論文誌、J62-A、12、pp.886-893 (1980)
- (4) Speech API Developer's Guide Version 1.0 Beta, Microsoft Corporation (1995)
- (5) Kraig Brockschmidt: INSIDE OLE2, Microsoft Press (1994)



太田 治徳 Harunori Ohta

1980 年入社。基本ソフトウェアの開発設計に従事。現在、青梅工場パソコンソフトウェア設計部課長。  
Ome Works



高橋 勉 Tsutomu Takahashi

1980 年入社。基本ソフトウェアの開発設計に従事。現在、青梅工場パソコンソフトウェア設計部。  
Ome Works



原 義幸 Yoshiyuki Hara

1980 年入社。音声合成システムの研究開発に従事。現在、マルチメディア技術研究所開発第六部。  
Multimedia Engineering Lab.