

# CELP 算法软件实现

刘永庆 沈兰荪

(北京工业大学信号与信息处理研究室, 100022)

**摘要** 介绍了 CELP 算法原理, 讨论了算法软件实现的有关问题, 并给出在 PC 机上的实现结果.

**关键词** 语音编码, 码激励线性预测, 语音合成

**分类号** TN912.326

## 0 引言

随着多媒体技术在计算机中的广泛应用, 具备音频硬件及相应音频功能的微机已十分普遍. 利用微机开展音频业务, 常需利用数据压缩技术来降低音频数据量, 比如要实现基于窄带信道的语音通信, 就必须进行语音压缩编码.

码激励线性预测 (Code Excited Linear Predictive coding, CELP)<sup>[1]</sup>是优秀的语音压缩编码算法. 它能在保持较高编码质量的前提下, 将原始语音数据 (8 kHz 采样) 压缩到 4.8 kbps 甚至更低. CELP 是当今中、低速率语音编码的主流技术之一, 在计算机中进行语音压缩编码就可以利用这种编码技术.

典型 CELP 编码具有较高的运算复杂度, 通常需用专门硬件 (比如 DSP) 才能实现. 然而随着通用 CPU 性能的提高, 利用 CPU 资源, 用纯软件方法实现 CELP 算法也是可能的. 实际上, 支持甚至集成音频编码功能是 CPU 技术发展的趋势. 比如 Intel 公司的 MMX 技术, 可增强 CPU 的多媒体能力, 对 IIR/FIR 滤波、线性预测分析以及码本搜索等语音编码常用技术的处理, 带 MMX 的 CPU 比普通 CPU 快 2~4 倍.

## 1 CELP 算法原理

CELP 算法基于语音的线性预测分析与合成原理, 用语音合成的方法生产语音, 其中语音合成包括两部分: 一是短时线性预测合成, 用来模拟声道响应, 形成语音谱包络. 二是长时相关合成, 用以模拟声门激励, 形成语音谱的谐波结构. 编码时, 先利用短时线性预测分析技术得到语音模型参数, 然后确定模型激励. 在典型 CELP 编码中, 模型激励采用随机噪声序列, 众多激励序列以码本形式同存于编解码双方, 模型激励的确定采用语音合成的方法搜索激励码本并依据合成语音听觉失真最小准则来确定合适的激励码矢.

CELP算法只对语音模型参数和模型激励进行编码. 解码时, 用激励码矢驱动语音合成模型即可得到解码语音.

典型 CELP 算法原理如图 1 所示:

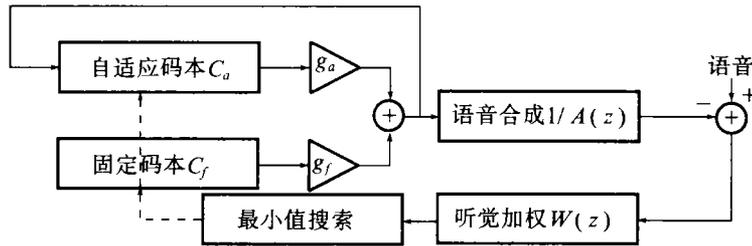


图1 CELP编码原理

其中语音合成滤波器  $1/A(z)$  用于语音短时相关合成; 自适应码本  $C_a$  及其增益  $g_a$  用于语音长时相关合成; 听觉加权滤波器  $W(z) = A(z)/A(z/r)$  用于改变编码噪声谱的形状, 从而利用人耳听觉特性掩蔽编码噪声; 固定码本  $C_f$  及相应的增益  $g_f$  为语音合成固有激励. 最佳的自适应及固定码矢可通过顺序搜索自适应码本与固定码本来得到<sup>[5]</sup>.

实际 CELP 编码出于算法简化或其它因素的考虑往往各具特色, 尤其是码本的构成有多种形式, 比如在 4.8 kbps 的 CELP 编码标准 FS 1016 中用的是叠接码本<sup>[2]</sup>, 而 8 kbps 的北美数字蜂窝电话语音编码标准 IS54 (改进 CELP 算法) 则采用了矢量和码本<sup>[3]</sup>.

当用软件方法实时实现 CELP 算法时, 限于 CPU 的运算能力, 要求算法应具有较低的复杂度; 此外, 对于实用系统, 语音采集与回放等基本音频功能的实现也必须考虑.

## 2 CELP 软件实现

### 2.1 CELP 算法简化

CELP 算法是一种非对称的编码算法, 编码复杂度远高于解码. 实际上, 不经过简化的 CELP 编码运算量高达每秒 100 M 次运算以上, 难以用软件方法实现, 所以必须对算法进行简化. 在前面提到的 FS1016 以及 IS54 是两种比较简捷且性能较好的 CELP 算法. 通过对它们进行算法仿真发现, 此两种算法具有软件实时实现的潜力. 为此, 我们以这两种算法为基础进一步简化算法, 开发了适于软件实现的低运算复杂度 CELP 算法.

低复杂度 CELP 编码原理如图 2 所示.

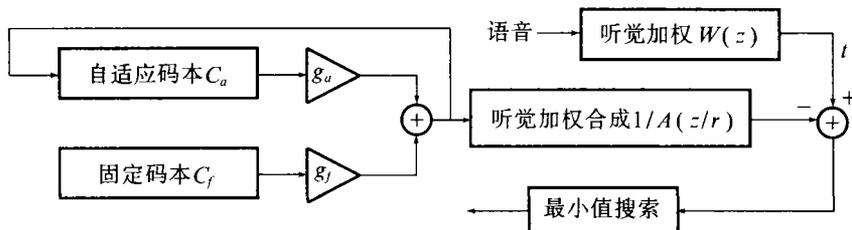


图2 CELP编码原理

算法简化思想主要有:

### 2.1.1 利用系统等效变换,降低运算复杂度

实际上,图2与图1在原理上是等效的,但其码本搜索运算量仅为前者的1/3.另外,对码本搜索是基于当前合成语音的零状态部分(典型做法则是利用全响应)与原始语音相比具有最小听觉加权失真,因此无需计算听觉加权合成的零输入响应<sup>[2]</sup>,实验表明,这种算法并不影响编码语音的质量.

### 2.1.2 固定码本搜索采用混合算法

具体方法是:用叠接码矢形式构成矢量和码本的基础矢量,从而用递推法快速计算基础矢量的卷积响应<sup>[3]</sup>.而在码本搜索中,则按矢量和码本的快速搜索法进行<sup>[3]</sup>.

### 2.1.3 采用多级预选法,降低码本搜索复杂度

标准 CELP 的误差准则是使合成语音与原始语音的听觉加权失真最小,按此准则搜索码本比较复杂,为此,我们采用了多级预选法降低码本搜索复杂度.这当中采用的预选准则是使合成语音在听觉意义上与原始语音相关最大.具体方法是计算码矢的听觉加权合成响应与目标矢量的相关值,并以此作为预选准则选出候选码矢.这当中,对自适应码本的搜索,可通过 backfiltering 法<sup>[4]</sup>来快速进行.而对于矢量和码本搜索,这一项的计算几乎毫不费力<sup>[3]</sup>.

接下来是按普通 CELP 误差准则计算用候选码矢合成语音的听觉加权误差,并从中选出最佳码矢.这当中,只需将候选码矢的序号排序,便可按某种递推方式进行误差计算,从而进一步提高运算效率<sup>[5]</sup>.

在 CELP 中,码本搜索是至关重要的,非最优的误差准则将导致编码性能的下降.在上述预选搜索法中,码本搜索的结果是非最佳的,然而由于误差中的相关项与误差大小有密切的联系,而且,利用语音的相关性作预选准则也符合听觉特性,因此,非最佳的码本搜索结果并不会对编码质量造成很坏影响.在实际编码实验中,预选法增大了编码的客观失真,但并不严重影响主观听觉质量.

### 2.1.4 灵活方式编码,降低码本搜索运算量

在 CELP 中,自适应码本和固定码本对语音合成的作用是不同的.一种观点认为:语音包括周期成分和随机成分,自适应码本主要用于周期成分合成,而固定码本主要用于随机成分合成.对于周期性强的语音段(浊音),过分的随机成分合成将导致语音质量下降,而对于随机性强的语音段(清音),则仅靠固定码本合成就能满足要求.为此,我们的做法是:对于自适应码本合成,只在周期性较强的语音段中才采用,而一旦起用自适应码本,同时将减小固定码本搜索范围,从而抑制其合成作用.显然,这样做将大大降低码本搜索运算量.至于语音段的判别,我们是基于码本预选中所采用的听觉加权相关项进行的.对于具有明显周期性的语音段,利用自适应码本所得的合成语音与原始语音是强相关的;反之,当语音具有随机特性时,则不存在这种相关性.为此,只需为相关项设定一阈值,即可用于分辨语音性质.

### 2.1.5 仔细调整编码参数,快速进行参数量化

调整编码参数可以在一定范围内调整编码的算法复杂度.为此,可在保证算法具有一定质量和满足量化速率的前提下,选择运行最快的参数.而对参数进行量化时,主要是采用运算简捷的量化方法,一般而言,标量量化比矢量量化运算简单,因而更适于用软件方

法实现。

### 2.1.6 注意编程技巧的使用, 提高程序执行效率

对此, 主要考虑两点: 首先, 算法实现细节需精益求精, 尽量做好算法的人为优化。比如, 对数组数据进行初始化或移位赋值时, 往往可利用内存块拷贝或块移动指令来加速实现。另外, 应尽量选用新的编程工具, 以充分发挥新一代 CPU 的卓越性能。比如, 当利用 Pentium CPU 来实现算法时, 就应选用支持 Pentium CPU 的编译系统来开发算法软件。

以上是算法简化的基本思路, 利用它对 CELP 算法进行简化, 将使运算复杂度大大降低, 从而便于软件实现。

## 2.2 参数调整及量化编码

CELP 编码速率由编码参数大致决定, 而编码参数的选择又与编码质量密切相关。对于图 2 所示的 CELP 编码, 在折衷考虑编码速率及编码质量后, 参数选取为: 语音分析帧长 30 ms (用于语音分析), 每一分析帧又分为 4 个编码帧 (用于激励确定), 线性预测分析为 10 阶 AR 模型, 自适应码本尺寸 128 (相应延迟为 60~187 样本点), 固定码本尺寸 128。

典型 CELP 中需要量化和编码的参数有: ①语音的谱参数。②基音周期 (自适应码矢序号) 及相应增益。③随机码矢序号及其增益。对这些参数的量化、编码应既满足一定的量化精度以保证高质量的编码, 同时, 又要尽可能做到量化编码的高效率以降低总的编码数据量。在 CELP 中, 当编码参数确定后, 最终编码速率是由量化、编码方案决定的。

CELP 各参数中, 比较容易编码的是激励参数。其中, 激励码矢 (固定、自适应码矢) 序号必须准确编码, 为此可直接用数码表示。而对激励增益的量化可将增益的极性和幅度分开进行。其中极性用单一比特表示, 这相当于在运算量不变情况下使码本尺寸等效增加 1 bit; 对增益幅值的量化, 可采用普通方法进行。比如用  $\mu$ 、A 率 PCM 或普通非均匀量化都能达到要求。实验表明, 对激励增益的量化大约需用 5 bit, 就可使量化前后的编码语音无听觉上的明显区别。

CELP 中最讲究的是语音谱参数的量化。这是因为粗略的语音谱参数量化将导致编码语音质量的显著下降。另外, 由于语音谱参数较多 (10 阶), 因此, 能否对谱参数进行高效编码是涉及整体编码效率的关键问题。

在对量化性能及算法复杂度进行综合考虑后, 我们选用偏向关系数的反正弦作为谱参数进行非均匀标量量化。

对于 4.8 kbps 的编码 (8 kHz 采样), 比特分配如附表所示:

实际编码结果表明, 按上述方法量化后, 不会造成语音质量的明显下降, 编码语音可保证清晰可懂, 且具有一定的自然度。

附表 一分析帧比特分配方案 (bit): 4.8 kbps

参 数	比特分析
谱参数 (10 阶)	44
自适应码矢序号	7×4
自适应码矢增益	5×4
固定码矢序号	7×4
固定码矢增益	5×4
语音判别	1×4
总计	144

## 2.3 基本音频功能实现

开发 CELP 的实时编码程序是基于普通 PC 机进行的。这当中, 除了语音编解码算法

外, 还要实现语音采集与回放等基本音频操作的功能. 实际上, 利用 Windows 系统操作平台, 只要配备声卡等音频硬件, 就可以方便地实现音频采集与回放等功能, 这一切完全由 Windows 内置多媒体功能所支持.

在 Windows 中涉及的音频数据主要有 3 种, 即波形音频、MIDI 音频和 CD-ROM 音频, 其中最为灵活的是波形, 我们也是利用波形来实现所需的语音功能.

Windows 中多媒体部分的结构如图 3 所示, 其中音频部分如图 4 所示.

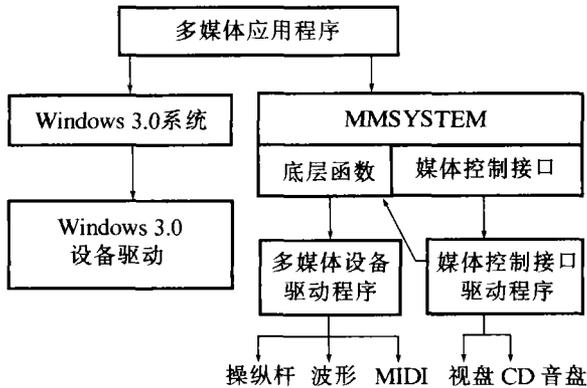


图3 多媒体扩展部分结构

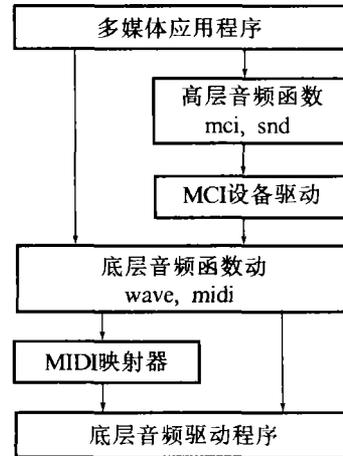


图4 多媒体音频编程原理

从图中我们可以看到, Windows 对波形音频的支持相当全面. 操纵波形既可通过高层音频函数或用 MCI, 也可通过底层音频函数进行. 其中, 底层音频函数具有最强的音频数据控制能力, 而这正是我们所需要的.

在进行波形的采集与回放中都存在的一个问题是要对波形数据进行管理, 即当波形数据输出或输入完毕后, 及时进行后续处理. 对此, 有标志查询法、窗口函数法和底层回调函数法 3 种方法. 利用底层音频函数构造实时语音处理程序的关键是要设置并管理好一个双缓冲结构的数据区. 其中, 缓冲区设置的容量要足够大, 以保证音频硬件填充一个缓冲区的时间大于音频数据的处理时间与系统调度所需时间之和.

以上是基于 Windows 操作平台进行音频采集、回放等基本功能及其用于实时音频处理的方法. 对于 CELP 软件实时语音编码软件的开发, 只需在窗口函数(回调函数)中进行实时编码即可. 由于采用了底层音频函数, 我们有能力在音频硬件允许范围内按任意质量采集语音, 然而这能否最终实现, 唯一的限制是编码算法的速度. 采用高采样率时会导致编码时间的增加, 且同时缩短数据缓冲区的填充时间, 从而使实时实现变得困难.

### 3 实验结果

利用前面所述的要点, 我们在 PC 机上实现了 4.8 kbps CELP 算法, 所用系统基本硬件配置为: Pentium 133 MHz CPU, 16 MB 内存, Sound Blaster 兼容声卡, 软件平台是中文 Windows 3.2, 编译系统为 Borland C/C++ 3.1. 算法实际运行表明, 每压缩 1 s 语音 (8

kHz 采样), 所需时间为 0.8~0.9 s. 而实际试听解码语音表明, 解码语音能保证基本的可懂度, 且音调自然, 因而质量良好.

## 4 小结

本文回顾了 CELP 算法原理, 讨论了 CELP 算法软件实现的有关问题, 并最终在通用 PC 机上予以实现. 其中还存在的问题主要有:

- 1) 编码质量和编码效率都有待提高.
- 2) 算法占用过多 CPU 资源, 严重影响其他辅助功能的实现.
- 3) 编程经验不足, 编程工具落后, 程序执行效率不高.

鉴于以上问题, 进一步的改进工作应着重以下方面:

首先, 在算法简化方面, 需作进一步努力. 另外, 应适当改善编码质量与效率等方面的性能. 最后, 在程序编制方面还应该作更细致的工作, 务求高性能算法最终能用程序高效率实现.

## 参 考 文 献

- 1 Schroeder M R, Atal B S. Code Excited Linear Prediction: High Quality Speech Coding at Very Low bit Rates, In: ICASSP. 1985. 937~940
- 2 Kleijn W B, Krasinski D J, Ketchum R H. Improved Speech Quality and Efficient Vector Quantization in SELP. In: ICASSP. 1988. 155~158
- 3 Geraon I A, Jasiuk M A. Vector Sum Excited Linear Prediction (VSELP) Speech Coding at 8kbps. In: ICASSP. 1990. 461~464
- 4 Adoul J P, Mabilieu P, Delprat M, Morissette S. Fast CELP Coding Base on Algebraic Codes. In: ICASSP. 1987. 1957~1960
- 5 刘永庆. 中、低速率语音压缩编码算法研究: [学位论文]. 北京: 北京工业大学电子工程学系, 1997

# The Software Realization of CELP

Liu Yongqing     Shen Lansun

( Signal and Information Processing Laboratory, Beijing Polytechnic University, 100022 )

**Abstract** The CELP speech coding algorithm is reviewed, some problems related to the software realization of CELP algorithm are discussed, and the algorithm proposed is realized on a PC as well.

**Keywords** speech coding, code excited linear predictive coder, speed sound synthesis