

低速语音编码中的预测分类分裂矢量量化技术*

马庆利¹, 季新生¹, 杨于村²

(1. 国家数字交换系统工程技术研究中心, 郑州 450002; 2. 华南理工大学 电信学院, 广州 510641)

摘 要: 为降低编码速率的同时仍能提供较好的谱失真性能, 提出了一种预测分类分裂矢量量化算法, 它根据线谱对的特点, 融合了预测、分类、分裂的方法对线谱对进行量化, 加入了记忆性。实验证明与其他几种方法相比, 该算法的量化性能在速率与失真间达到了较好的平衡, 且计算量大大降低, 仅占有内存有所增加。

关键词: 语音编码; 线谱对; 多级矢量量化; 预测分类分裂矢量量化; 谱失真

中图分类号: TN912.3 文献标志码: A 文章编号: 1001-3695(2009)10-3700-03

doi:10.3969/j.issn.1001-3695.2009.10.028

Predictive switched split vector quantiser of very low bit-rate speech coder

MA Qing-li¹, JI Xin-sheng¹, YANG Yu-cun²

(1. National Digital Switching System Engineering & Technological Research Center, Zhengzhou 450002, China; 2. School of Electronic & Information Engineering, South China University of Technology, Guangzhou 510641, China)

Abstract: In order to reduce bit-rate and still maintain fine distortion performance, this paper proposed predictive switched split vector quantization method. It added memory according the characteristic of the LSP. And it was a hybrid of switched split vector quantization techniques. Experimental results show that the PSSVQ provides a better trade-off between bit-rate and distortion performance than the others. In addition, the PSSVQ has a lower computational (search) complexity at the expense of an increase in memory requirements.

Key words: speech coding; line spectral pair(LSP); multi-stage vector quantization(MSVQ); predictive switched split vector quantization; spectral distortion

在强人为干扰下或环境噪声极强条件下的军用通信、数字语音保密通信中,难以新建或扩展信道,在这种条件下,极低速率语音编码就成为惟一选择。为了在降低语音编码速率的同时保证语音质量,就需要高效率的参数量化方法。线谱对由于良好的量化特性和内插特性以及可以保证合成滤波器的稳定性而广泛用于各种编码系统中。

矢量量化器研究的目标主要是围绕降低速率、减小失真、降低复杂度展开的。在固定速率下,直接用全搜索矢量量化器,则减小失真的问题得以解决。但由于其计算复杂度、码书占用内存大而备受限制,要对编码进行结构限制,一般的结构限制的码矢量量化方法为乘积码矢量量化,包括多级矢量量化、分裂矢量量化(split vector quantization,SVQ)等。多级矢量量化主要是对各级残差进行量化;分裂矢量量化是把矢量分成几个子矢量,然后对子矢量应用独立的码本进行量化;分类分裂矢量量化(switched split vector quantization,SSVQ)相比分裂矢量量化提高了率失真性能,计算复杂度也得以降低,近年受到追捧。文献[1,2]将多级矢量与SSVQ强制相结合且是无记忆的,性能其实有失真性,不过仍可以利用其部分思想对SSVQ进行优化。

上面几种量化都是无记忆的,由于LSP参数的有序性,且相邻参数间隔必须大于50 Hz,LSP参数在帧内与帧间都有相关性。预测矢量量化由于加入记忆的功能,其输入矢量和重构矢量的误差等于差值矢量的量化误差^[3],加入预测性比直接

对矢量量化时的量化误差要小,其量化特性更好一些。采用预测的方式对LSP预测残差进行矢量量化,可以进一步消除LSP参数的相关性,更进一步降低了码率。G.723.1^[4]运用了AR预测分裂矢量量化方法;由3GPP制定的AMR语音编码标准也运用了一阶滑动平均(MA)预测分裂矢量量化。G.729^[5]运用了预测两级矢量量化,在第二级运用分裂矢量量化。

本文运用分类分裂矢量量化结合预测矢量量化的特性对量化方法进行改进,以较少的比特数使LSP矢量达到了透明量化效果,且量化时间复杂度小,而码本训练尽管复杂,由于是离线操作,不影响量化编码器性能,把这种量化方法称为预测分类分裂矢量量化(PSSVQ)。

1 PSSVQ 算法设计

1.1 编码

本方案采用的预测分类分裂矢量量化是在预测矢量量化^[4,6]和分类、分裂矢量基础上融合而成的。其方案如图1所示。

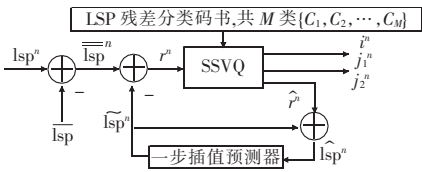


图 1 PSSVQ 编码方案

1) 线性预测

收稿日期: 2009-01-12; 修回日期: 2009-03-02 基金项目: 国家“863”计划资助项目(2007AA012431)

作者简介: 马庆利(1982-),男,山西朔州人,研究生,主要研究方向为语音编码、移动通信(maqingli19820904@126.com);季新生(1968-),男,江苏人,教授,博导,主要研究方向为移动通信、无线安全;杨于村(1970-),女,江西人,博士研究生,主要研究方向为移动通信系统及安全技术。

图 1 中 lsp^n 为 lsp 第 n 帧参数,参数维数为 $P=10$, $\overline{\text{lsp}}$ 为由训练数据统计过的 lsp 参数的均值矢量,所以去均值后的 lsp 值为

$$\overline{\text{lsp}}_i^n = \text{lsp}_i^n - \overline{\text{lsp}}_i \quad (i=1,2,\cdots,10) \tag{1}$$

设 $\hat{\text{lsp}}_i^n$ 为 $\overline{\text{lsp}}_i^n$ 的量化值,如果考虑到 lsp 的帧内和帧间相关性,用一步预测器对 $\overline{\text{lsp}}_i^n$ 预测,值为 lsp_i^n 如式(2):

$$\text{lsp}_i^n = a_i \hat{\text{lsp}}_{i-1}^n + b_i \hat{\text{lsp}}_{i-1}^{n-1} \quad (i=1,2,\cdots,10) \tag{2}$$

其中: a_i 为帧内预测系数; b_i 为帧间预测系数; a_i 和 b_i 值可以通过求均方误差最小来求得

$$E_i = \sum_{n=1}^N [r_i^n]^2 = \sum_{n=1}^N w_i [\overline{\text{lsp}}_i^n - \text{lsp}_i^n]^2 \tag{3}$$

其中: w_i 是 Paliwal-Atal 权^[7]; N 为训练数据总帧数;通过 E_i 对 a_i 和 b_i 分别求偏导数并令其为 0,可求得 a_i 、 b_i 。

文献[8]给出了如式(4)来计算 10 维线谱对频率参数的归一化相关系数

$$R[i,j,k] = \frac{\sum_{m=0}^{N-k-1} v_i[m] v_j[m+k]}{\sqrt{\sum_{m=0}^{N-k-1} (v_i[m])^2 \sum_{m=0}^{N-k-1} (v_j[m+k])^2}} \tag{4}$$

$$(i,j=1,2,\cdots,10;k=0,1,2,\cdots)$$

其中: v 是去均值后的线谱对频率参数; N 是用来计算归一化相关系数的全体样本矢量的个数,相关测度是在某一帧的第 i 个 lsp 和前面第 k 个帧的第 j 个 lsp 间进行的。因此, $k=0$ 时,可以用来计算 LSP 帧内相关系数; $k \neq 0$ 时,可以计算 LSP 帧间相关系数。本文采用的是一阶预测,所以 $k=1$;为研究计算方便,本文先只考虑其帧间相关性,即 $i=j$ 的情况,则 $a_i=0$;因此式(4)变为

$$R[i] = \frac{\sum_{m=0}^{N-2} v_i[m] v_i[m+1]}{\sqrt{\sum_{m=0}^{N-2} (v_i[m])^2 \sum_{m=0}^{N-2} (v_i[m+1])^2}} \quad (i=1,2,\cdots,10) \tag{5}$$

式(2)变为式(6):

$$\text{lsp}_i^n = b_i \hat{\text{lsp}}_{i-1}^{n-1} \tag{6}$$

根据式(5)运用 Levinson-Durbin 很简单地可以求出预测系数,因为它只是一个 2×2 的矩阵运算。进一步可求出残差:

$$r_i^n = \overline{\text{lsp}}_i^n - \text{lsp}_i^n \quad (i=1,2,\cdots,10) \tag{7}$$

2) 分类分裂矢量量化 (SSVQ)

求出的 r_i^n 进入图 2 所示分类分裂矢量量化系统中进行量化。具体如下:

先进行分类的选择,判断 r_i^n 是属于 M 类中的哪类码本;然后在对应类中码本查找,查找前先把矢量 r^n 分成两部分,前四维构成一矢量,后六维构成一矢量(SVQ 研究表明 4~6 分裂具有最佳性能^[8]),再分别在其对应的码本中找到合适的码字。

输出参数索引号为 i^n 、 j_1^n 、 j_2^n 。 i^n 代表分类的方向码本号; j_1^n 、 j_2^n 为在所分类的码本中找到的分裂码本的码字索引号。

1.2 译码

图 3 为 PSSVQ 的解码框图。

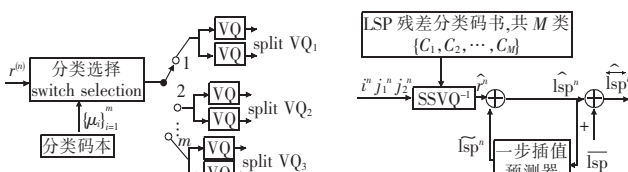


图 2 SSVQ 编码过程

译码过程具体如下:

a)接收端收到编码的索引号 i^n 、 j_1^n 、 j_2^n ,先根据 i^n 判断属于 M 个类中的哪个类,然后在相应类中再根据 j_1^n 、 j_2^n 分别在不同码书中查找对应码字,如由 j_1^n 查找到码字向量为 $\{\text{lsp}_1, \text{lsp}_2, \text{lsp}_3, \text{lsp}_4\}$,由 j_2^n 查找到的码字矢量为 $\{\text{lsp}_5, \text{lsp}_6, \text{lsp}_7, \text{lsp}_8, \text{lsp}_9, \text{lsp}_{10}\}$,则由这两个码合成构成一个 10 维的向量 $r^n = \{\text{lsp}_1, \text{lsp}_2, \cdots, \text{lsp}_{10}\}$ 。

b)通过 SSVQ⁻¹ 系统查找完 r^n 后再求 $\text{lsp}^n = \text{lsp} + r^n$ 。其中 lsp 是 lsp^n 一步插值预测的结果。图中的一步插值预测器已定。实际上,对 lsp^n 的求解相当于 r^n 通过一个滤波器,这个滤波器由一步插值预测器所决定。

c)因为最初的编码分析是对去均值后的 lsp 进行的,解码后的 lsp 值要加上均值。

$$\hat{\text{lsp}}^n = \overline{\text{lsp}} + \hat{\text{lsp}}^n \tag{8}$$

其中: $\hat{\text{lsp}}^n$ 为最终解码结果; $\overline{\text{lsp}}$ 为由训练矢量确定的 lsp 固定均值。

1.3 码书设计

a)设训练矢量共 N 个,表示为 lsp^n ($n=1,2,\cdots,N$)。每个 lsp^n 是 10 维矢量,先求出整个训练矢量每维的平均值,即

$$\overline{\text{lsp}}_i = (1/N) \sum_{n=1}^N \text{lsp}_i^n \quad (i=1,2,\cdots,10) \tag{9}$$

其中: lsp_i^n 为第 n 帧训练矢量的第 i 个分量。

b)应用式(1)把所有训练矢量每维的均值分离出来,即

$$\overline{\text{lsp}}_i^n = \text{lsp}_i^n - \overline{\text{lsp}}_i \quad (n=1,2,\cdots,N; i=1,2,\cdots,10) \tag{10}$$

这样训练矢量值明显变小,动态范围变小,易得到精确的码本。

c)应用式(6)求出每个训练矢量的预测值 lsp_i^n

d)应用式(7)求出残差 r_i^n ($n=1,2,\cdots,N; i=1,2,\cdots,10$)。这 N 个残差矢量就是最终的训练矢量,运用 lbq 算法先把这个训练矢量集归类到 M 个胸腔,同时求出每个胸腔的质心。

e)对 M 个胸腔中某一类的训练矢量再单独训练码本。方法是:先将每个训练矢量分成两个子矢量,前 4 个为低端矢量 $r_{i_1}^n$ ($i_1=1,2,3,4$);后 6 个为高端矢量 $r_{i_2}^n$ ($i_2=5,6,\cdots,10$)。分别对高端和低端训练矢量单独训练码本。

f)对 M 个胸腔中的其他胸腔类用同样的方法进行码本的训练。

2 编码性能测试

运用谱失真测度进行编码的效果测试,需要把经过量化的 LSP 转换为 LPC ,再与未经量化的 LSP 对应的 LPC 谱相比较得出结论。对第 i 帧的谱失真定义为

$$\text{SD}_i = (1/\pi) \int_0^\pi (20 \lg |H_i(w)| / |H_i(\hat{w})|)^2 dw \tag{11}$$

求积运用数值分析^[9] 中高斯公式的复化梯形法,即

$$\int_a^b f(x) dx \approx h \sum_{n=0}^{N-1} f[a + (n+1/2)h] \tag{12}$$

所以,

$$\text{SD}_i = (20/\pi) \sqrt{\Delta w \sum_{n=0}^{L-1} \{ \lg [H(w_n + \Delta w/2)] - \lg [\hat{H}(w_n + \Delta w/2)] \}^2} \tag{13}$$

其中: L 为计算点数, $w_n = (\pi - 0) n/L = \pi n/L$, $\Delta w = \pi/L$;这里 $H(w)$ 和 $\hat{H}(w)$ 分别为输入、输出语音的短时 LPC 谱,即有

图 3 PSSVQ 译码过程

$$H^{(i)}(w_n) = 1/|A^{(i)}[\exp(-jwn)]|^2$$
$$\hat{H}^{(i)}(w_n) = 1/|\hat{A}^{(i)}[\exp(-jwn)]|^2 \tag{14}$$

$A^{(i)}[\exp(-jwn)]$ 和 $\hat{A}^{(i)}[\exp(-jwn)]$ 是第*i*帧语音未量化LPC和量化的LPC多项式。

LPC达到透明量化的指标是:在频0~3 000 Hz间的平均谱失真接近1 dB,平均谱失真在2~4 dB的范围内的比例为2%左右,平均谱失真大于4 dB的比例趋于0。

本文对PSSVQ算法与MSVQ、SVQ、SSVQ作比较。它们的时间复杂度和内存需求如下^[10]:

$$\text{complexity}_{\text{MSVQ}} = \sum_{i=1}^p (4n \times 2^{b_i} - 1)$$
$$\text{memory}_{\text{MSVQ}} = \sum_{i=1}^p n \times 2^{b_i}$$
$$\text{complexity}_{\text{SVQ}} = \sum_{i=1}^s (4n_i \times 2^{b_i} - 1)$$
$$\text{memory}_{\text{SVQ}} = \sum_{i=1}^s n_i \times 2^{b_i}$$
$$\text{complexity}_{\text{SSVQ}} = (4n_i \times 2^{b_m} - 1) + \sum_{i=1}^s (4n_i \times 2^{b_i} - 1)$$
$$\text{memory}_{\text{SSVQ}} = n \times 2^{b_m} + 2^{b_m} \sum_{i=1}^s n_i \times 2^{b_i} \tag{15}$$

其中:*b_m*是代表分类分裂矢量量化器(SSVQ)中分类方向的比特数,如分类成16个方向,则*b_m*=4。*p*是级数;*s*是分裂数;*n*是总的维数,本文对10维lsp分析,*n*=10。*b_i*和*n_i*:*b_i*是子码本个数(多级矢量量化指每一级向量个数,分裂法指分裂后每一类的码向量数)对应的比特数,*n_i*是由*b_i*代表码向量的维数。

对于所占内存数不难理解,即为所有码本里码字个数与其对应维数的积,然后再求和。而对时间复杂度的求法,目前主要有两种方法^[6]:a)每量化一个输入矢量需要乘法运算的次数;b)每量化一个输入矢量需要加(减)法、乘法和比较运算的次数,本文采用b)方法。以SVQ为例,complexity_{SVQ} = $\sum_{i=1}^s (4n_i \times 2^{b_i} - 1)$,分裂共*s*个子矢量,对每个子矢量,假设输入子矢量为*X*,码字为*Y_i*,进行比较要作*K*次相减、*K*次相乘、*K*-1次相加。现在输入矢量*X*要与*N*个码字进行比较,就需要*K*×*N*次相减、*K*×*N*次相乘和(*K*-1)×*N*次相加。为了搜索到一个与最小矢量对应的码字,还需要作*N*-1次比较运算,因此,时间复杂度为*b*=*KN*+*KN*+(*K*-1)*N*+*N*-1=3*NK*-1。*N*和*K*即这里的*z^{b_i}*和*n_i*。而对于分裂成*s*个子矢量来说,显然是要加求和号。即有complexity_{SVQ} = $\sum_{i=1}^s (3n_i \times 2^{b_i} - 1)$,这个是无权重的SVQ复杂度,有权重就有complexity_{SVQ} = $\sum_{i=1}^s (4n_i \times 2^{b_i} - 1)$,多加了*NK*次的乘法运算。

PSSVQ是在SSVQ基础上加了预测性,其内存与SSVQ法相当(只增加了一个10维的LSP均值矢量,可忽略)。时间复杂度,只增加了*n*次相乘,2*n*次相减,与SSVQ相比,也可完全忽略。表1~4为不同编码方案的性能比较。其中谱失真(平均谱失真,谱失真在2~4 dB所占百分比,谱失真大于4 dB所占百分比)是实验所得,复杂度和占用内存是计算所得。

表 1 MSVQ 谱失真、计算复杂度及内存需求

每帧的 比特数	均谱失真/ dB	2~4/ dB/%	>4/ dB/%	复杂度/ kflops/帧	占用内存/ floats
for <i>p</i> = 2					
24(12+12)	0.950	1.50	0	327.678	81 920
23(12+11)	0.987	1.81	0.005	245.758	61 440
22(11+11)	1.061	2.02	0.007	163.838	40 960
21(11+10)	1.167	2.31	0.012	122.878	30 720

表 2 SVQ 谱失真、计算复杂度及内存需求

每帧的 比特数	均谱失真/ dB	2~4/ dB/%	>4/ dB/%	复杂度/ kflops/帧	占用内存/ floats
for split(4~6)					
24(12+12)	1.211	1.67	0.006	163.838	40 960
23(11+12)	1.325	2.34	0.013	147.454	32 768
22(11+11)	1.471	3.59	0.019	81.918	20 480
21(10+11)	1.629	5.02	0.024	65.534	16 384

表 3 SSVQ 谱失真、计算复杂度及内存需求

每帧的 比特数	均谱失真/ dB	2~4/ dB/%	>4/ dB/%	复杂度/ kflops/帧	占用内存/ floats
for <i>bm</i> = 3, split(4~6)					
24(3+10+11)	0.893	0.23	0	65.853	131 152
23(3+10+10)	0.963	0.56	0	41.277	82 000
22(3+9+10)	1.001	1.06	0.007	33.085	65 616
21(3+9+9)	1.134	1.25	0.024	20.797	41 040

表 4 PSSVQ 谱失真、计算复杂度及内存需求

每帧的 比特数	均谱失真/ dB	2~4/ dB/%	>4/ dB/%	复杂度/ kflops/帧	占用内存/ floats
for <i>bm</i> = 3, split(4~6)					
24(3+10+11)	0.801	0.04	0	65.883	131 162
23(3+10+10)	0.873	0.23	0	41.307	82 010
22(3+9+10)	0.924	0.56	0.006	33.115	65 626
21(3+9+9)	0.983	1.24	0.010	20.827	41 050

3 结 束 语

实验证明,本文所提出的PSSVQ方法,在21 bit就达到了透明量化的效果,在所比较的四种方法中,对相同比特数进行量化,PSSVQ的量化性能也最好,而其计算复杂度非常小,仅内存占有量有所增加。下一步工作可以再对帧内相关性加入预测,且运用多步插值预测量化效果将更好,而为降低复杂度或内存量可以在分类、分裂的级数上做工作。

参考文献:

[1] RAM M S S, SIDDAlAH P, LATHA M M. Multi switched split vector quantizer[J]. International Journal of Computer, Information, and Systems Science, and Engineering, 2008, 2(1): 1-6.

[2] RAM M S S, SIDDAlAH P, LATHA M M. Multi switched split vector quantization of narrowband speech signals[C]//Proc of World Academy of Science, Engineering and Technology. 2008: 236-239.

[3] 孙圣和, 陆哲明. 矢量量化技术及应用[M]. 北京: 科学出版社, 2002: 116-142.

[4] ITU-T Dual Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kbps[S]. Recommendation G. 723. 1, 1996.

[5] ITU-T Recommendation G. 729, Coding of Speech at 8 kbps Using Conjugate- structure Algebraic-code-excited Linear-prediction (CS-ACELP)[S]. 1996.

[6] 鲍长春. 数字语音编码原理[M]. 西安: 西安电子科技大学出版社, 2007: 135-137, 161-164.

[7] PALIWAL K K, ATAL B S. Efficient vector quantization of LPC parameters at 24 bit/frame[J]. IEEE Trans on of Speech and Audio Processing, 1993, 1(1): 3-14.

[8] CHU W C. Speech coding algorithms-foundation and evolution of standardized coders[M]. New York: Wiley Inter Science, 2003: 398-403.

[9] 王能超. 数值分析简明教程[M]. 北京: 高等教育出版社, 1984: 89-96.

[10] SO S, PALIWAL K K. Efficient product code vector quantisation using the switched splitvector quantiser[J]. Digital Signal Processing, 2007, 17(1): 138-171.