

文章编号:1001-9081(2010)03-0761-04

基于语音质量预测的 Speex 自适应码率控制算法

蔡铁¹, 龙志军², 伍星¹

(1. 深圳信息职业技术学院 信息技术研究所, 广东 深圳 518029;

2. 中兴通讯股份有限公司 UMTS 基带部, 广东 深圳 518057)

(cai.tie@163.com)

摘要:为实现 IP 语音 (VoIP) 质量的动态管理与控制, 提出了一种基于语音质量预测的自适应码率控制算法。通过实时预测 VoIP 通话的瞬时语音质量和总体语音质量, 自适应地调整 Speex 编码参数, 从而根据需要选择最佳编码速率。实验仿真结果表明, 提出的算法能够有效减少网络拥塞, 提高 VoIP 系统的语音质量。

关键词: IP 语音; 语音编码; Speex; 自适应码率控制; 语音质量预测

中图分类号: TP393 **文献标志码:** A

Adaptive rate control algorithm of Speex codec based on speech quality prediction

CAI Tie¹, LONG Zhi-jun², WU Xing¹

(1. Institute of Information Technology, Shenzhen Institute of Information Technology, Shenzhen Guangdong 518029, China;

2. UMTS Baseband Department, ZTE Corporation, Shenzhen Guangdong 518057, China)

Abstract: To manage and control the speech quality of VoIP communication dynamically, an adaptive rate control algorithm based on speech quality prediction was proposed. This algorithm predicted the instantaneous speech quality and the integral speech quality of VoIP in real time to adjust the encoding parameters of Speex codec adaptively. Then it selected the optimal encoding bit rate in need. The simulation results demonstrate that the proposed algorithm indeed reduces the network congestion and improves the speech quality of VoIP system.

Key words: VoIP; speech coding; Speex; adaptive rate control; speech quality prediction

0 引言

IP 语音 (Voice over IP, VoIP) 是近年来广泛应用的一种基于 IP 网络的实时语音通信系统, 具有通话成本低廉、占用带宽小、能提供各种多媒体服务等优点, 是当今计算机网络和通信领域研究的热点。为了支持 Internet 上的 VoIP 应用, 需要尽量减小延迟、丢包等网络损伤对语音通信质量的影响, 在最大化网络资源使用率、避免网络拥塞的条件下获得更高的语音通信质量。因此, 近年来已有一些研究者在 VoIP 应用的码率和质量控制方面开展了大量的研究工作。例如文献[1]采用网络抖动作为网络质量的参数来控制自适应多码率 (Adaptive Multi-Rate, AMR) 编解码器的码率。文献[2]则采用丢包率作为 VoIP 系统中 ARM 编解码器码率选择的判决参数。文献[3]提出了一种码率自适应方法, 通过计算延迟和丢包的滑动平均阈值实现码率选择。但是, 上述这些方法所采用的判决参数如延迟、丢包等都不能正确、直接地反映网络真实状况, 限制了码率控制方法的性能。文献[4]采用语音质量感知评价 (Perceptual Evaluation of Speech Quality, PESQ) 算法计算语音质量的平均意见得分 (Mean Opinion Score, MOS) 值, 用于改变 AMR 语音编码的码率, 获得了一定的效果。但这种算法每 5 s 需要对比接收端语音信号和参考语音信号以计算语音质量的 MOS 值, 因此缺乏灵活性和实时性。

文献[5]采用对 G. 711、G. 726 和 G. 729 三种速率的语音编码算法进行动态选择, 有效地提高了 VoIP 语音通话的质量。但所提方法只适用于国际电信联盟 (International Telecommunication Union, ITU) 的语音编解码算法, 对于 Speex、iLBC 等新算法或非 ITU 标准算法将无法应用。针对上述算法存在的问题, 本文提出了一种新的基于语音质量预测的自适应码率控制算法用于 Speex 语音编解码器, 该算法能够实时地预测 VoIP 语音质量用于控制 Speex 编码参数, 从而有效减少网络拥塞, 提高 VoIP 语音质量。

1 Speex 语音编解码器

Speex 是一种开源的基于码激励—线性预测 (Code Excited Linear Prediction, CELP) 的可变速率语音编解码器^[6], 它通过对原始语音帧进行线性预测编码 (Linear Prediction Coding, LPC) 分析, 得到产生这组语音数据的声道模型参数, 从而实现高质量和低比特率的编码。Speex 不仅提供了基于码激励—线性预测算法的编解码模块, 而且还提供了噪声消除、静音抑制和自动增益控制等语音预处理模块和声学回声消除模块, 为保障 IP 网络中的语音通信质量提供了技术手段。此外, Speex 还支持多种比特率, 如 8 KHz 采样的低比特率 (窄带 2.15 ~ 24.6 Kbps)、16 KHz 采样的中比特率 (宽带 3.95 ~ 42.2 Kbps) 以及 32 KHz 采样的高比特率 (超宽带), 并

收稿日期: 2009-09-29; 修回日期: 2009-11-11。

基金项目: 广东省自然科学基金资助项目 (8151802904000012); 深圳市科技计划项目 (SZKJ0708)。

作者简介: 蔡铁 (1977-), 男, 湖南长沙人, 副教授, 博士, 主要研究方向: 信号处理、语音处理与识别、模式识别; 龙志军 (1976-), 男, 湖南长沙人, 工程师, 硕士, 主要研究方向: 宽带无线通信、信源编码、信道编码; 伍星 (1980-), 女, 湖南娄底人, 讲师, 硕士, 主要研究方向: 网络信息系统、信息安全。

具有可变比特率 (Variable Bitrate, VBR) 的特性, 可在任意时刻动态地改变语音的比特率, 特别适用于 Internet 上的语音通信。

Speex 编码主要受到其编码质量参数 Q (取值范围 0 ~ 10) 的控制^[6], 质量参数的取值越大, 编码速率越高。Speex 窄带语音编码质量参数与编码速率的关系如表 1 所示。在本文所提算法中, Speex 的编码质量参数 Q 将被设置为不同的整数值, 从而获得不同的编码速率。

表 1 Speex 窄带语音编码质量参数与编码速率关系表

编码质量参数	编码速率/Kbps	编码质量参数	编码速率/Kbps
0	2.15	6	11.00
1	3.95	7	15.00
2	5.95	8	15.00
3	8.00	9	18.20
4	8.00	10	24.60
5	11.00		

2 自适应码率控制算法

对于 VoIP 系统而言, 基于发送端的自适应码率控制能通过选择最佳的编码策略以匹配当前的网络状态, 从而提高一段时间内的语音流质量。已有的自适应码率控制算法大多采用网络延迟、丢包等参数作为自适应调整的依据, 但已有的研究工作表明, 语音质量预测能够综合考虑端到端延迟、丢包以及编码器特点等因素, 从而准确地判断出 VoIP 通信状态, 有利于提高自适应码率控制算法的性能。

2.1 语音质量预测

IP 网络的语音质量客观测量方法可以分为侵入式和非侵入式两类。其中侵入式测量方法具有更高的准确度, 但这种方法在测量时需要使用参考数据和网络, 因此一般不适用于在线实时测量。ITU-T P. 862 定义的 PESQ 方法^[7]是当前一种广泛应用于 VoIP 系统的侵入式语音质量测量方法, 它需要对比失真语音信号和参考语音信号来计算语音质量的 MOS 值, 同时也没有考虑 IP 网络延时、抖动等因素对语音质量测量的影响。非侵入式测量方法不需要参考信号, 可以利用网络相关参数 (例如丢包率、延时、抖动和编解码器等) 或失真语音信号直接预测语音通话质量。ITU-T E-model^[8]是一种非侵入式语音质量测量的计算模型, 它直接利用网络相关参数进行计算, 但目前 E-model 只能用于有限的几种编码器和网络环境, 对于其他一些研究未涉及或新的参数组合情况还处于实验研究阶段; 而 ITU-T P. 563^[9]则通过分析失真语音信号来估计 MOS 分值, 它必须经过多次测量并对结果进行平均才能得到较可信的测量结果, 因此这种方法并不适用于测量个别呼叫。本文算法采用 PESQ 和 E-model 相结合的非侵入式语音质量测量方法^[10]来预测网络语音通话质量, 从而自适应地控制 Speex Codec 的编码速率。

2.2 非侵入式语音质量测量方法

1) 根据当前丢包率和 Speex 码率获得样本语音的 PESQ MOS 分值。

2) 将 PESQ 的 MOS 值转换为 R 值:

$$R = 3.026MOS^3 - 25.314MOS^2 + 87.060MOS - 57.336 \quad (1)$$

PESQ 的 MOS 值没有考虑延迟的影响, 因此其 R 值没有将延迟损伤 I_d 计算在内, 则丢包损伤 I_{e-eff} 为:

$$I_{e-eff} = R_0 - R \quad (2)$$

其中 $R_0 = 93.2$ 。

3) 计算延迟损伤 I_d :

$$I_d = 0.024d + 0.11(d - 177.3)H(d - 177.3) \quad (3)$$

其中: $x < 0$ 时, $H(x) = 0$; $x \geq 0$ 时, $H(x) = 1$ 。

式(3)在延迟 400 ms 内比较准确, 但超过 400 ms 则有较大误差, 因此可采用更精确、更复杂的公式计算 I_d :

$$I_d = -2.468 \times 10^{-14}d^6 + 5.062 \times 10^{-11}d^5 - 3.903 \times 10^{-8}d^4 + 1.344 \times 10^{-5}d^3 - 0.001802d^2 + 0.103d - 0.1698 \quad (4)$$

4) 计算 E-Model 的 R 值:

$$R = R_0 - I_d - I_{e-eff} \quad (5)$$

5) 计算 E-Model 的 MOS 值:

$$MOS = \begin{cases} 1, & R \leq 0 \\ 1 + 0.035R + R(R - 60) \times \\ \quad (100 - R) \times 7 \times 10^{-6}, & 0 < R < 100 \\ 4.5, & R \geq 100 \end{cases} \quad (6)$$

2.3 瞬时语音质量

每次语音通话都由语音段和静音段组成, 每个语音段称为一个语音会话突峰 (talkspurt), 静音段处于两个语音会话突峰之间。在本文所提的自适应码率控制算法中, 上述非侵入式语音质量测量方法将用于测量每个语音会话突峰的语音质量, 作为瞬时语音质量。但由于背景流量的突发特性, 两个连续语音会话突峰的瞬时语音质量可能差别很大。对于自适应码率控制算法而言, 该算法关注的是一段时间内的语音流质量, 它不仅需要测量瞬时通话质量, 而且要测量会话过程中一段时间内的总体语音质量。

2.4 总体语音质量

总体语音质量可以通过计算会话过程中一段时间内的瞬时语音质量的平均值来获得, 但这种平均值的计算方法过于简单, 存在较大的误差, 因此本文采用了文献[11]提出的计算方法, 充分考虑了人耳的听觉感知特性。

根据文献[11], 可将一次通话 (Call) 看作若干 8 s 区间段组成的序列, 每个区间段的语音质量为区间段内瞬时语音质量的平均值 (即区间段内所有语音会话突峰语音质量的平均值), 通话的总体语音质量则为所有区间段语音质量的加权。因此, 总体语音质量的 MOS 值 MOS_t 可按式(7)进行计算:

$$MOS_t = \frac{\sum_i W_i \cdot MOS_i}{\sum_i W_i} \quad (7)$$

其中: MOS_i 为第 i 段语音的平均 MOS 值; W_i 为第 i 段语音的相对权重。 W_i 的计算公式为:

$$W_i = \max [1, 1 + (0.038 + 1.3 \times L_i^{0.68}) \times (4.3 - MOS_i)^{0.96 + 0.61 \times L_i^{1.2}}] \quad (8)$$

其中 L_i 为第 i 段语音的位置, 其取值为 0 到 1, 每次通话的开始处 $L_i = 0$, 通话的结束处 $L_i = 1$ 。

2.5 自适应码率控制算法

基于发送端的码率控制, 需要将网络状态或语音质量报告给发送端, 这个过程一般采用实时控制协议 (Real-time Control Protocol, RTCP) 实现, 因为 RTCP 的报文中包含了网络丢包和延时变化的统计信息, 这些报文一般每 5 s 就周期性地发送一次^[4]。但是对于码率控制而言, 采用 RTCP 的控制方法过于缓慢, 难以针对网络状态的变化实时地作出反应。为了提高码率控制的反应速度, 可以缩短 RTCP 报文的发送周期, 但同时也会大幅增加网络流量。与传统方法不同, 本文算法采用按需控制的模式, 当网络状态变化需要发送端参数进行调整时, 算法才向发送端发送控制信息。

除了编码速率 (码率) 以外, 包长 (packet size) 也是重要

的语音编码参数,对于 VoIP 通信质量有着直接的影响。文献[5]的实验结果表明,增加包长会提高端到端的延迟,但同时也会降低每次通话的 IP 码率,因此增加包长能够提高拥塞网络中的语音通话质量。尤其当链路中的语音流量比例较高时,改变包长能获得明显的质量改善。在本文提出的自适应码率控制算法中,包长的调整将作为码率调整的辅助措施,从而更好地控制 VoIP 语音通话质量。

本文算法的描述如下。

1) 在抖动缓冲前采集 IP 包延时的统计信息。

2) 计算包播放时间(packet playout time)并进行自适应抖动缓冲。

3) 计算每个语音会话突峰(talkspurt)的语音质量,即瞬时语音质量:

① 计算当前语音会话突峰内的包丢失率;

② 测量端到端的延时,它在一个语音会话突峰内为固定值;

③ 由丢包率获得 PESQ 的 MOS 分值,然后按 PESQ 和 E-model 相结合的非侵入式语音质量测量方法计算出瞬时语音质量,记为 Q_i 。

4) 在丢包率为零、网络延时最小的情况下,计算当前编码速率所能获得的最高语音质量 Q_{\max} 。

5) 计算总体语音质量 MOS_T ,记为 Q_T 。

6) 码率控制的判决。码率控制的判决主要参照的是总体语音质量和瞬时语音质量,分为以下三种情况。

① 当 $Q_{\max} - Q_T > Th1$ 时,总体语音质量 Q_T 很低,此时应立即调整编码参数:

If $Q_{\max} - Q_i > Th3$,此时瞬时语音质量 Q_i 也很低,必须大幅度调整编码参数,因此选择最低的码率,并设置包长为 30 ms;

If $Q_{\max} - Q_i \leq Th3$,此时瞬时语音质量 Q_i 较高,只需小幅调整编码参数,因此保持当前码率不变,在包长大于 10 ms 的情况下减小包长 10 ms;

② 当 $Th2 < Q_{\max} - Q_T \leq Th1$ 时,总体语音质量 Q_T 较低,此时应根据情况适当调整编码参数:

If $Q_{\max} - Q_i \leq Th3$,此时瞬时语音质量 Q_i 较高,只需小幅调整编码参数,因此在包长大于 10 ms 的情况下保持当前码率不变、减小包长 10 ms,在包长等于 10 ms 的情况下适当提高码率,并设置包长为 30 ms;

If $Th3 < Q_{\max} - Q_i \leq Th4$,此时瞬时语音质量 Q_i 也较低,只需小幅调整编码参数,因此保持当前码率不变,在包长小于 30 ms 的情况下增加包长 10 ms;

If $Q_{\max} - Q_i > Th4$,此时瞬时语音质量 Q_i 很低,必须较大幅度调整编码参数,因此降低码率,并设置包长为 10 ms;

③ 当 $Q_{\max} - Q_T \leq Th2$ 时,总体语音质量 Q_T 较高,此时应根据情况适当调整编码参数:

If $Q_{\max} - Q_i \leq Th3$,此时瞬时语音质量 Q_i 较高,网络状况良好,可小幅调整编码参数,因此在包长大于 10 ms 的情况下保持当前码率不变、减小包长 10 ms,在包长等于 10 ms 的情况下适当提高码率,并设置包长为 30 ms;

If $Th3 < Q_{\max} - Q_i \leq Th4$,此时瞬时语音质量 Q_i 较低,可能只是暂时的网络性能下降,因此保持当前码率和包长不变;

If $Q_{\max} - Q_i > Th4$,此时瞬时语音质量 Q_i 很低,必须调整编码参数,因此保持当前码率不变,在包长小于 30 ms 的情况下增加包长 10 ms;

7) 码率控制的执行。为了保证系统语音质量的稳定,防止语音质量起伏较大,自适应码率控制的执行必须延迟一定

时间,即等待当前编码参数已执行后才进行新的调整。一般情况下,本文算法在等待一个语音会话突峰的时间后才执行码率控制。

本文算法采用了三种不同速率的 Speex 窄带编码: 24.6 Kbps(参数 $Q = 10$,压缩比为 2.6:1)、15 Kbps(参数 $Q = 8$,压缩比为 4.27:1)和 8 Kbps(参数 $Q = 4$,压缩比为 8:1),选择初始编码速率为 24.6 Kbps。与文献[4]的研究相似,本文算法对于总体语音质量的判断依据是 $Th1$ 和 $Th2$ 两个阈值。 $Th1$ 取值为 0.5,表示已可感知到明显的总体语音质量下降; $Th2$ 取值为 0.2,表示总体语音质量下降不明显,需避免不必要的调整。瞬时语音质量主要受到抖动缓冲引起的延迟和丢包两个因素的影响,其中端到端延迟对瞬时语音质量的影响相对较小,250 ms 延迟引起的质量下降仅为 0.3 MOS 至 0.4 MOS 左右,而突发的丢包则对瞬时语音质量的影响较大,一个会话语音突峰中 30 个包丢失仅 3 个包,就会导致瞬时语音质量下降约 0.9 MOS。因此,本文算法对于瞬时语音质量的判断依据同样也采用两个阈值: $Th3 = 0.3$ 和 $Th4 = 1$,当 $Q_{\max} - Q_i$ 的值超过阈值 $Th4$ 时,表示瞬时语音质量很低,而低于阈值 $Th3$ 时,则表示瞬时语音质量较高。

3 实验仿真与分析

实验采用 VoIP 网络仿真对本文算法进行性能研究,并与未采用自适应码率控制算法的仿真结果进行了对比分析。仿真采用 Speex 窄带语音编码,所用的网络结构如图 1 所示,其链路容量设置为 5 Mbps,网络时延假设为 100 ms 固定值,链路流量由语音流和作为背景流量的数据流组成。当网络中链路使用率较低时,采用最高速率的 Speex 窄带语音编码可获得最佳的 VoIP 语音通话质量,这种情况下无需采用自适应码率控制算法;当网络中链路使用率较高时(链路使用率超过 80%),通过瓶颈路由器的突发背景流量可能导致网络拥塞,此时调整编码速率将有效提高 VoIP 语音质量。因此,本文的实验仿真将针对网络拥塞(即链路使用率较高)的情况,对自适应码率控制算法进行测试与分析。

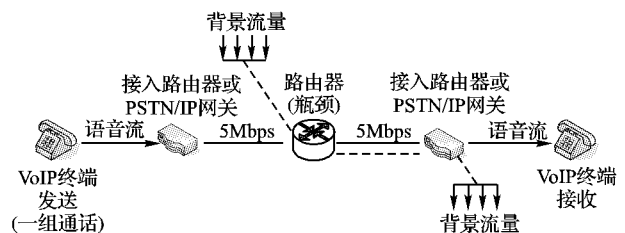


图1 仿真网络结构

实验仿真结果如图2~3、表2所示,仿真采用了一组VoIP通话,所有通话能够同时控制以采用相同的编码参数,并假设每个通话具有相同的质量变化。每个VoIP通话包括100个语音会话突峰,每个语音会话突峰为300 ms,即通话时长为30 s。仿真所用的语音流比率为 $V = \text{语音流量} / \text{链路容量} = 0.6$,数据流比率为 $D = \text{数据流量} / \text{链路容量} = 0.4$,链路使用率 $U = V + D = 1$,平均数据流量(即平均背景流量应用)为2 Mbps。图2是不采用自适应码率控制算法的仿真结果,此时VoIP通话采用固定码率Speex算法($Q = 10$, 24.6 Kbps),网络拥塞明显,VoIP通话出现较多的延迟和丢包,语音质量较低,平均感知MOS值(即总体语音质量MOS的平均值)仅为3.029。图3则是采用本文自适应码率控制算法的仿真结果,其仿真条件与图2情况完全相同。从图3可以看出,VoIP语音质量得到显著提高,平均感知MOS值达到了3.471。

图 2 与图 3 仿真结果的通话质量特征参数对比如表 2 所示,表中平均 MOS 为瞬时语音质量 MOS 的平均值,平均感知 MOS 则为总体语音质量 MOS 的平均值。从表 2 可以看出,采用本文的自适应码率控制算法将有效降低延时和丢包,提高 VoIP 语音质量。

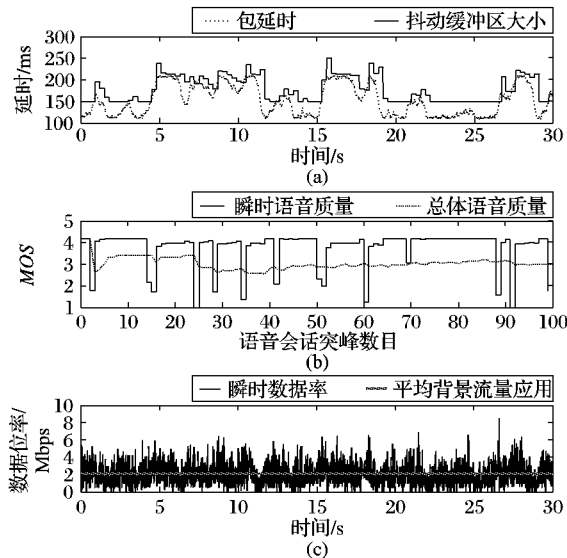


图 2 不采用自适应码率控制算法的 VoIP 语音质量

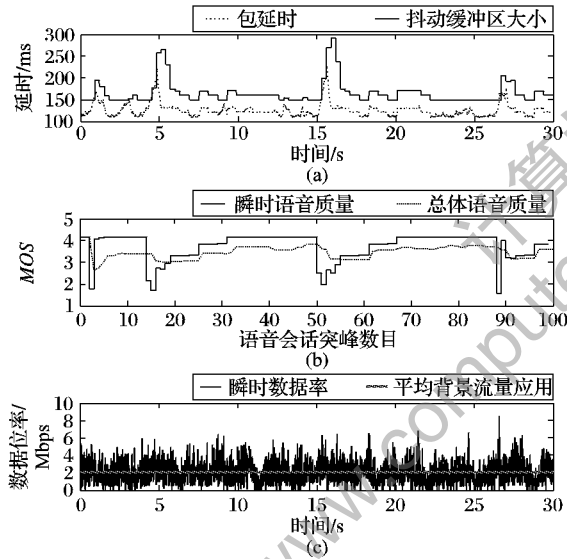


图 3 采用自适应码率控制算法的 VoIP 语音质量

表 2 仿真结果的通话质量参数

码率控制	平均延 时/ms	语音丢 包率/%	数据丢 包率%	平均 MOS	平均感 知 MOS
不采用自适应码率 控制(固定码率)	179.09	5.8	3.2	3.032	3.029
采用自适应码率 控制(本文算法)	165.69	1.8	0.1	3.408	3.471

在不同链路应用的情况下,本文针对所提出的自适应码率控制算法作了进一步的实验仿真,并与不采用本文算法的实验结果进行了对比。实验仿真结果如表 3 所示,它是每种链路应用情况分别进行 10 次实验的平均值。从表 3 可以看出,链路使用率越高,自适应码率控制算法的效果越明显。而在一定的链路使用率的情况下,不同的语音流和数据流比率可能导致较大差别的 VoIP 语音质量,语音流比率越低,数据流引起网络拥塞的可能性越高,VoIP 语音质量的 MOS 分值就越小,此时自适应码率控制算法的效果越明显。表 3 的仿

真结果表明,本文提出的自适应码率控制算法可以有效提高 VoIP 系统中 Speex 语音编解码算法的性能,获得更高的 VoIP 语音质量。

表 3 不同链路应用情况下的仿真结果

语音流 比率 V	数据流 比率 D	链路使 用率 U	平均 MOS	
			不采用本文算法	采用本文算法
0.3	0.5	0.8	4.169	4.169
0.5	0.3	0.8	4.170	4.170
0.3	0.6	0.9	3.362	3.505
0.4	0.5	0.9	3.497	3.600
0.6	0.3	0.9	4.145	4.151
0.4	0.6	1.0	2.920	3.260
0.6	0.4	1.0	3.151	3.423
0.8	0.2	1.0	3.120	3.605

4 结语

针对 VoIP 系统中语音编码速率的控制问题,本文提出了一种新的基于语音质量预测的自适应码率控制算法用于 Speex 语音编解码器,该算法能够实时地预测 VoIP 语音质量用于控制 Speex 编码参数,从而有效减少网络拥塞,提高 VoIP 语音质量。本文算法不仅可以用于 Speex 编解码算法,而且能够广泛应用于 ITU 或非 ITU 的各种多码率语音编解码算法,同时也能将多种语音编解码算法混合使用,自适应地选择一种最佳码率的编解码算法。

参考文献:

[1] SEO J W, WOO S J, BAE K S. A study on the application of an AMR speech codec to VoIP [C]// Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Salt Lake City, UT, USA: IEEE Computer Society, 2001: 1373-1376.

[2] ABREU-SERNANDEZ V, GARCIA-MATEO C. Adaptive multi-rate speech codec for VoIP transmission [J]. Electronic Letters, 2000, 36(23): 1978-1980.

[3] NGAMWONGWATTANA B. Sync & sense enabled adaptive packetization VoIP [D]. Pittsburgh: University of Pittsburgh, 2007.

[4] QIAO Z, SUN L, HEILEMANN N, et al. A new method for VoIP quality of service control use combined adaptive sender rate and priority marking [C]// Proceedings of the 2004 IEEE International Conference on Communications. New Jersey: IEEE Computer Society, 2004: 1473-1477.

[5] MYAKOTNYKH E. Adaptive speech quality in voice-over-IP communications [D]. Pittsburgh: University of Pittsburgh, 2008.

[6] VALIN J M. The Speex codec manual [EB/OL]. [2009-06-20]. <http://www.speex.org/docs/>.

[7] International Telecommunication Union. Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs [S], 2001.

[8] International Telecommunication Union. The E-model, a computational model for use in transmission planning [S], 2005.

[9] International Telecommunication Union. Single-ended method for objective speech quality assessment in narrow-band telephony applications [S], 2004.

[10] SUN LING-FEN, IFEACHOR E C. Voice quality prediction models and their application in VoIP networks [J]. IEEE Transactions on Multimedia, 2006, 8(4): 809-820.

[11] ROSENBLUTH J H. Testing the quality of connections having time varying impairment [S], 1998.