

Klatt 合成器定点实时化及其在汉语 文语转换中的应用

许 可^{1,2)} 谢咏圭¹⁾

(1) 北京师范大学无线电电子学系, 100875, 北京; (2) 中国科学院空间科学与应用研究中心, 100080, 北京;

第一作者 27 岁, 男, 硕士)

摘要 通过对 Klatt 合成器通道浮点范围的研究和实验, 确定了采用单双精度整数混合使用的定点算法来模拟 Klatt 合成器的浮点运算, 以便在 TMS 320C25 上实现实时合成, 同时对原合成器在 6 个方面进行了改进. 改进后的合成器一方面满足了实时合成的要求, 实时合成的结果基本上达到了原算法的精度和动态范围; 另一方面为汉语规则合成提供了更为有效的手段. 文语转换系统的硬件部分由一台 386 微机 and 一块 TMS 320C25-D 型高速信号处理板组成的主从式系统构成, 系统中考虑了汉语中的声调规则、轻声规则等 5 种韵律规则, 提高了系统的自然度和可懂度.

关键词 共振峰; 实时; 语音合成; 文语转换

分类号 TN 912.3

语音合成离不开合成器. Klatt 串并联共振峰合成器是美国 MIT 著名的言语和听力学家 D. Klatt 的研究成果, 他于 1980 年在美国声学杂志上发表了《串并联共振峰合成器合成软件》^[1]. 这个软件普及到世界各国许多语音实验室, 在国际上享有盛誉的 MITTALK 和 DECTALK 二文语转换系统就是在这个软件基础上开发的^[2]. 1988 年 Klatt 对原软件进行了重大改进, 并命名为 KLSYN88, 由于 Klatt 合成器在英语文语转换中的成功应用, 所以采用该合成器进行汉语的文语转换是本文工作的出发点.

目前, 已有一些学者从汉语文语转换的角度对 Klatt 合成器的最新版本 KLSYN 88 合成器进行了研究, 从中得出 KLSYN 88 对汉语的文语转换来说是比较理想的参数合成器.

1 Klatt 合成器定点算法的研究及其改进措施

用 Klatt 合成器进行汉语的文语转换, 需要借助于高速信号处理器进行实时合成, 此前中国科学院声学所与北京师范大学合作在 TMS 320C30 高速信号处理板上实现了实时合成^[3,4], 但由于该板成本太高, 不利于推广应用, 所以在寻求廉价成本的时候, TMS 320C25 就成为一种较为理想的选择.

同 C25 相比, C30 最大的优点是具有浮点运算的功能, 一个浮点乘运算在一个指令周期内即可完成 (一个 C30 的指令周期为 60 ns), 所以 C30 在大量浮点数据处理方面具有得天独厚的优势; 相比之下 C25 是定点处理器, 其指令周期为 100 ns^[5,6]. 由于 Klatt 合成器中的绝大多数运

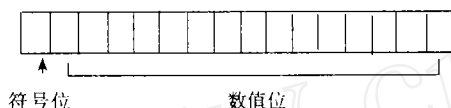
* 国家“八五”重点攻关课题资助项目
收稿日期: 1994-11-29

算都是浮点运算, 所以用 C25 来进行实时合成, 要解决的问题有 2 个: 一是用定点算法来模拟浮点运算, 二是进行算法优化.

用定点算法来代替 Klatt 合成软件中的浮点运算又要求达到实时是本文的一个难点. Klatt 合成器汉语音节合成的采样率是 $10 \text{ K} \cdot \text{s}^{-1}$, 由于其指令周期为 100 ns , 所以要实时合成, 就要求在 1000 条指令内合成一个样点. 经估算用双精度整数和调用浮点运算子程序都不可能达到实时要求, 所以要寻找一种算法简单而又能保证精度的定点算法.

1.1 浮点问题的定点处理 本文在浮点数处理上采用浮点近似归整法, 通过对声源和滤波系数用浮点近似归整法归整, 再用单双精度整数混合算法, 在精度和运算时间上达到最佳的折衷, 实现了实时合成, 不过也带来了一定的误差, 浮点和定点实时合成的语图不完全一致, 但误差较小, 在听觉上没有察觉出它们之间的差异.

下面给出单精度的浮点近似归整法的一般原理:



在 16 bit 位中第 15 位是符号位, 第 14 ~ 0 位是数值位. 一个浮点数被整型化后的数值的大小由下式决定:

整型数值 = (浮点数 × 幅度因子) 的整数部分,

其中的幅度因子一般由使用者自行设定, 其原则是使整型数值尽可能的利用 15 个 bit 位, 这样误差最小. 同样, 对于双精度的浮点近似归整法, 32 bit 位中第 31 位是符号位, 第 30 ~ 0 位是数值位.

1.1.1 对通道滤波器输入输出范围的分析 对于声道的全极点模型, 其传输函数为 $V(Z) = G / (1 - \sum_{k=1}^N a_k Z^{-k})$, 式中 N 是极点个数, G 是幅值因子, a_k 是系数, k 是正整数^[7]. 此时可将此传输函数分解为多个二阶极点的网络的串联, 则得:

$$V(Z) = \prod_{i=1}^M \frac{a_i}{1 - b_i Z^{-1} - c_i Z^{-2}}, \quad (1)$$

式中 $c_i = -\exp(-2\pi B_i T)$, $b_i = 2\exp(-\pi B_i T) \cos(2\pi F_i T)$, $a_i = 1 - b_i - c_i$, $G = a_1 \cdot a_2 \cdot a_3 \cdots a_M$, M 是小于 $(N+1)/2$ 的整数, T 是取样周期. 取式 (1) 中的某一级

$$V_i(Z) = a_i / (1 - b_i Z^{-1} - c_i Z^{-2}), \quad (2)$$

令 $Z = e^{j\omega}$, 则可得出其幅频特性如下

$$|V_i(e^{j\omega})| = |a_i| / \{ [1 - b_i \cos(\omega) - c_i \cos(2\omega)]^2 + [b_i \sin(\omega) + c_i \sin(2\omega)]^2 \}^{1/2}.$$

Klatt 合成器中的各滤波器都具有 (2) 式所示的幅频特性. 为了研究该合成器中各滤波器的输入输出的量化位, 下面对汉语语音图中的 3 个角元音 [a]、[i] 和 [u] 及央元音 [e] 的幅频特性进行比较, 实验中采用的 4 个元音的共振峰频率是用阻抗相移法计算得出的结果 (表 1)^[6].

通过 ILS 软件包的分析可知, 不同元音同一共振峰的带宽相差不大, 实验中采用该软件包的分析结果, 5 级共振峰的带宽分别为 60、90、150、200、200 Hz. 由 (2) 式计算得到各滤波器幅频的最大增益如表 2.

从计算结果可以看出, 元音在各谐振点都有较大的增益, 最大的达 45.8 dB. 由于 Klatt 自然

表1 各元音的共振峰频率

Hz

元音	F_1	F_2	F_3	F_4	F_5
a	621	1263	2596	3428	4204
i	219	2222	3311	3907	4740
u	304	827	2614	3756	4710
e	467	1806	2895	3423	4687

表2 各滤波器幅频的最大增益

dB

元音	F_1	F_2	F_3	F_4	F_5
a	20.4	23.4	27.1	29.4	35.9
i	11.4	29.4	31.2	32.9	45.8
u	14.2	19.5	27.2	31.7	44.8
e	17.8	27.0	28.7	29.4	44.2

声源最大输出为 1 920 000, 所以串联通道某些谐振器在一些谐振点上的输出超过单精度整数范围 ($-32\ 768 \sim 32\ 767$), 而没有超过双精度整数的范围 ($-2\ 147\ 483\ 648 \sim 2\ 147\ 483\ 647$), 所以串联通道的输入输出在不压缩的情况下, 必须用双精度整数来表示。

对于并联通道, 由于噪音声源最大的输出可达 8 191, 清音各谐振器的幅频最大增益不超过 30 dB (由于汉语对清音共振峰的研究较少, 实验中对用 ILS 软件包提取的汉语音节中的清声部分作过统计分析, 其幅频最大增益都在 30 dB 以下), 且 Klatt 合成器对并联通道各谐振器的幅度压缩都在 20 倍以上, 所以该通道各滤波器的输入输出不超过单精度整数范围 ($-32\ 768 \sim 32\ 767$), 即可以用单精度整数来表示。

1.1.2 实验测试结果 在实验中采用汉语音节是“阿”、“依”、“乌”、“哦”、“司”、“杀”、“帕”、“开”、“哗”, 这几个音包含了元音图中的 3 个角元音、央元音以及汉语中有代表性的辅音, 所以对用这 9 个音节的参数合成的声音进行统计分析, 基本上可以确定 Klatt 合成器中各浮点变量的动态范围, 实验中还采用了 3 个分别由男声、女声和童声发音的“今天下雨有点冷”的句子。

对于串联声道, 各级谐振器的最大输出数值是 237 538, 超过单精度整数范围 ($-32\ 768 \sim 32\ 767$), 所以串联通道的各级谐振器的输入输出采用 32 位的双精度整数表示。对于并联通道, 各级谐振器的最大输出值是 4 362, 所以对并联通道的各级输入输出采用 16 位的单精度整数表示。

作者也试图用 16 位单精度整数表示串联通道各滤波器的输入输出, 即将该通道的输入值压缩至 16 位, 可是合成声音误差较大且动态范围较小, 故该方法不可行。Klatt 合成器中有大量的乘法运算, 所需时间为

- 一个 16×16 位乘法 …… 1 条 C25 指令 (单精度整数乘法)
- 一个 16×32 位乘法 …… 20 条 C25 指令 (单双精度整数乘法)
- 一个 32×32 位乘法 …… 50 条 C25 指令 (双精度整数乘法)

本文采用单双精度整数混合算法, 一方面满足精度要求, 另一方面较为省时。由于该算法完成一次乘运算需 20 条指令, 这样合成一个样点约需 1 600 条 C25 指令, 不满足实时要求, 所以必须对原合成器进行改进, 改进后合成一个样点约需 700 条 C25 指令, 可以达到实时合成。

1.2 Klatt 合成器的改进

1.2.1 设置通道开关 由于喉声源并联通道一般不常用, 所以原模型在用喉声源合成时走串联通道, 合成摩擦音走并联通道, $A_v + A_h$ 和 A_f 分别控制 2 个声源的振幅。要关闭一个通道时, 即令其中的一个值为零, 这样合成一个样点实际上通过了 2 个通道的运算, 只是一个通道的输出是零。

改进后的模型比以前模型增加了一个开关——CP2 (如图 1 所示), 使得大多数情况下合成一个样点实际上只通过一个通道 (当 $A_v + A_h$ 和 A_f 都不为零时例外), 这样减少了运算时间。当 $A_v + A_h$ 或 A_f 由零变为非零值时, 打开对应通道; 当 $A_v + A_h$ 或 A_f 由非零值变为零时, 不马上关闭该

通道, 延迟一个语音帧之后再关闭, 这样处理是为了保留该通道滤波器的残留效应, 以便和原合成器的输出一致。

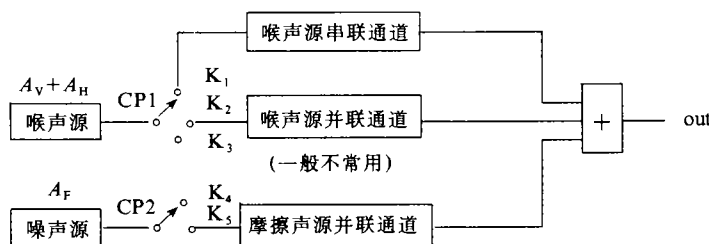


图 1 改进后的 KLSYN88 简略框图

1.2.2 平滑算法的改进 由于不同音节的共振峰参数是不一样的, 为了消除 2 个音节拼接时由于共振峰参数的突变而带来的噪声, KLSYN88 设计了平滑算法。以第 3 共振峰的平滑为例, 原算法为:

$$\begin{aligned} a_3 &= (\text{float}) F_3 / a_3, \\ r_{3c-1} &= r_{3c-1} \times a_3, \\ r_{3c-2} &= r_{3c-2} \times a_3, \\ a_3 &= F_3, \end{aligned}$$

其中 F_3 是当前音节第 3 共振峰频率, a_3 是前一音节第 3 共振峰的频率值, r_{3c-1} , r_{3c-2} 是第 3 共振峰谐振器的当前输出和上一次输出。由于该算法是浮点运算, 如果直接用于定点运算就失去了平滑作用。改进后的定点算法如下:

$$\begin{aligned} r_{3c-1} &= r_{3c-1} \times (F_3 - F_{31}) / F_{31} + r_{3c-1}; \\ r_{3c-2} &= r_{3c-2} \times (F_3 - F_{31}) / F_{31} + r_{3c-2}; \\ F_{31} &= F_3; \end{aligned}$$

改进后算法采用了原算法的思想, 并体现了整型量的平滑作用。

1.2.3 参数刷新程序的改进 Klatt 合成器算法中, 合成一个点需 60 个参数, 这当中有的是不变参数, 有的是每一帧都变化的参数。原算法中对每一帧都开辟了一个 60 个参数的单元空间, 这样内存开销较大。改进的算法中建立 3 张参数表, 即基本参数表、动态参数索引表和动态参数表。基本参数表中开辟 60 个参数单元空间; 动态参数索引表含有基本参数表中要被刷新的参数的数目及位置; 动态参数表只含有每帧都变化的参数。在合成时, 用基本参数表中的 60 个参数合成当前帧的数据, 合成一帧后, 根据动态索引表, 把下一帧变化的参数从动态参数表中提取出来, 刷新基本参数表, 这样合成完一帧后, 基本参数表被刷新一次, 下一帧再根据已被刷新过的基本参数表来合成语音。

以“今天下雨有点冷”为例, 原算法需要开辟 12K 字的数据区存放参数, 而改进算法只需要 2.3K 字的数据区空间。

1.2.4 对 D_0 , U_1 参数的改进 D_0 是 Klatt 合成器的第 1 个参数, 它代表被合成语音的长度 (单位是 ms), 改进算法将该参数的作用改为合成声音的总的帧数, 以使算法简便。 U_1 是 Klatt 合成器的第 2 个参数, 它代表每帧合成时间的长度 (单位是 ms), 这是一个固定参数。实验中将 U_1 参数变为可变参数, 即每一帧合成的时间可长可短。改变的依据是语音信号是随时间变化的短时稳定信号, 随时间变化表现为不同语音帧有不同的参数; 短时稳定表现为在同一语音帧内, 即在

一小段时间内语音参数不变, 把 U_i 变为可变参数, 只是使得每个短时稳定的语音帧的长短发生变化, 只要取样率保持不变, 除了时长因素外, 语音的其他特征不变。

改变 U_i 参数后, 可以将一句话拉长和缩短, 这一方法为汉语的文语转换调节韵律特征提供了十分有效的手段。汉语中音节时长是一个重要的韵律特征, 例如轻声、多字词中音长的分配、强调语气等等都需要调节音节时长, 以往的作法通常采用抽取或插入帧的办法, 即抽取一些语音帧参数使语音信号变短, 插入一些语音帧参数使语音信号变长。这种传统作法有 2 点不利因素: 一是算法较为复杂, 且控制不灵活; 二是抽取和插值后, 语音参数平滑程度不够, 特别是时长变化较大时有可能由于抽取和插值后的参数不满足滤波器的稳定条件从而产生“咯啦”声。

把 U_i 改为变量后, 弥补了传统方法的不足, 不但控制灵活, 而且不改变其他任何参数, 不会引入任何误差和噪声, 在本文中将该方法运用到汉语文语转换系统中, 提高了系统的性能。

1.2.5 自适应系统的建立及动态范围变化的比较 由于定点化的要求, 整个通道所用的 bit 位是一定的, 这给系统的动态范围带来了限制, 当信号源幅度变化时, 有可能引起个别滤波器的输出越界而产生振荡, 所以对信号源的幅度应加以限制。

本系统中, 应用 C25-D 型开发板 12 位输出, 而原 Klatt 合成器软件输出时用 16 位, 因而要对输出结果压缩 16 倍, 由此产生了 2 种方法: 一种是直接对输出结果进行压缩, 另一种是对声源压缩。对声源压缩可以使整个通道有较大的动态范围, 但对输出音质有影响; 对输出结果进行压缩可以保证有较好的音质, 但通道的动态范围变小, 而音质和动态范围都是汉语文语转换系统的非常重要的指标, 考虑到这两方面因素的影响, 本文采用声源幅度自适应的调节方法。由于辅音通道的动态范围较小, 故该方法只用于浊音通道。

例如若浊音幅度因子的全程增益控制 $G_v < 55$ dB, 则声源幅值压缩 32 倍; 若 $65 \text{ dB} < G_v < 70$ dB, 则声源幅值压缩 8 倍, 表 3 给出了自适应算法与直接对结果进行压缩的动态范围变化。由于该算法是针对串联通道, 所以测试实验中选用的音节含有汉语元音图中的 3 个角元音 [a]、[i] 和 [u], 另外选用一个含有清音成分的音节“杀”。从测试的结果上来看, 声源的自适应算法基本上达到了原浮点算法的动态范围。

1.2.6 指数、余弦函数的查表运算方法

指数和余弦函数都是浮点函数, 如果直接用定点算法进行浮点模拟运算则计算量很大, 针对指数和余弦函数所取参数的特点, 本文在允许的误差范围内采用查表法以简化算法。

需要说明的是原 KLSYN 88 合成器的设计

表 3 动态范围对比表

测试音节	原浮点算法 (动态范围)	直接对结果压 缩的定点算法	声源自适应 定点算法
“阿”	30 ~ 70	36 ~ 65	30 ~ 70
“依”	45 ~ 80	50 ~ 75	45 ~ 78
“乌”	45 ~ 80	48 ~ 75	45 ~ 80
“杀”	50 ~ 75	55 ~ 70	50 ~ 74

是非常完美的, 本文中所作的一些改进措施主要是针对用 TMS 320C 25 来开发实时合成器的需要来进行的。原算法更多地从合成器信号处理的完善性角度来考虑, 对一些诸如运算时间和存储器的开销一类的因素考虑较少, 然而这些因素正是本文工作面临的难点, 本文所做的改进措施的大部分是用来解决这些问题的, 但对 U_i 参数的改进可以说是对 Klatt 合成器设计思想的重大改进, 该措施使得 Klatt 合成器在规则合成方面具有更大的灵活性和开发潜力。

2 Klatt 合成器的实时化研究

语音的实时合成要求在一个取样周期内合成一个点, 为了达到这一目的, 通常采用 2 种方

法:一种是算法优化,使合成一个点的时间尽可能的短,这一步在前面的叙述中已介绍;另一种方法是采用高速信号处理器,即在一个取样周期内完成更多的运算指令,本文中采用 TMS 320C25 来实现实时合成算法。

2.1 C25 的内存组织 C25 有 64K 字的程序存储器和数据存储器的寻址范围,在实验中把片内的 544K 字的 RAM 映射成数据存储器;同时设置 $M_p/M_c=1$,使 C25 片内 ROM 占有的程序存储器低地址的 4K 字映射为外部程序存储器。

1) 程序存储器 程序存储器前 32 个单元用来存放中断向量和保留位,在程序中利用 TINT 内部定时器中断进行放音,主程序从程序存储器的 32 号单元存放,第 0 号存储单元,即外部复位信号中断向量填写一条跳转指令,指向 32 号单元开始执行主程序。

2) 数据存储器 数据存储器前 6 个单元为存储器映象寄存器单元,程序中通过对周期寄存器、定时器寄存器、中断寄存器和全程存储器的设置,使内部定时器的中断频率为 10 kHz,合成数据通过串行口数据发送寄存器送至 D/A 转换器。其中在片内第 0 页放有 6 个变量,第 6、第 7 页放有 192 个变量,故共存有 198 个数据变量,因 0~7 页为片内存储器,存取速度快,所以将变量放于片内有利于提高运算速度。

在片外	400h ~ 43dh	存放基本参数表,
	480h ~ 493h	存放动态参数索引表,
	494h ~ 27FFh	存放动态参数表,
	2800h ~ 7800h	存放语音合成数据。

3) 数据区的循环寻址 2800h ~ 7800h 为语音合成数据区,算法上采用循环方式寻址,即 2800h ~ 7800h 地址首尾相连,合成语音数据在环形地址上寻址,这样大大地节省了数据存储器的开销。由于 C25-D 型板只提供了 32K 字的数据存储器,所以循环寻址方式为在该板上合成长时间的语音提供了技术手段。

2.2 Klatt 合成器的实时实现 实时系统的硬件是由一台 PC/386 机与 TMS 320C25-D 型开发板构成的主从式系统构成的,主机负责从音库中读取语音参数文件,建立 3 张参数表,并向 C25 加载合成程序和 3 张参数表,启动 C25 工作;从机 C25 负责语音的实时合成。

主机与从机间的通信是通过公用存储器资源并使用 I/O 口传送控制信息来实现的,在 C25 运行程序的同时,主机可以做其他工作,例如对下一段将要合成的语音进行参数处理等,从而体现着主从机的并行操作。主机也可以通过 &H325 口和 &H32B 口向 C25 输入输出数据,并且,由此可以监测 C25 的运行状态;C25 也可以通过 0 号口向主机发送数据,以实现信息的相互传递。

2.3 实验结果 本文把用 C25 定点实时合成的结果与原 KLSYN88 浮点运算的结果做一比较,并将比较结果做在语图上。实验中选取了元音图中的 3 个角元音 [a]、[i]、[u] 及央元音 [e]。实验中还比较了一个句子:“今天下雨有点冷”。从语图的比较来看,2 种合成结果并不完全一致,原因在于定点算法中采用了浮点近似归整法来近似表示浮点数,这种方法在原理上就存在着一定的系统误差,但是误差很小,不会引起听觉的变化。2 种方法合成的声音在实际听辨中,从效果上来看是一致的。所以在汉语语音合成中,用 C25 定点算法实现的 Klatt 合成器完全可以达到原浮点合成器的听觉效果,并且大大降低了合成器的硬件成本,因此有利于进一步的推广应用。

3 基于 TMS 320C25 的实时汉语文语转换系统的初步实现

文语转换系统是一种高级的语音合成系统, 它将文字材料按规则转换成流畅的语音, 一个完整的文语转换系统需要一个音库、一个规则库和一个实时合成器。语音合成器采用基于 TMS 320C25 的 Klatt 合成器, 下面介绍语音库和规则库。

1) 语音库的建立 系统中使用的语音库是用 ILS 软件包已经分析好了的 418 个无音调音节的共振峰参数库。经过 ILS 软件的分析可以获得 Klatt 合成器的控制参数, 并将它写成合成器要求的 *.DOC 文件的形式。

2) 语音规则库 现规则库中收录了 5 条规则, 分别是 4 级重音模式, 句末语调模式, 多字词中音长的前长模式, 声调规则, 轻声规则。

3) 实时实现 本文实现的汉语文语转换系统是由一台 386 微机 and 一块 TMS 320C25-D 型高速信号处理板组成的主从式系统构成的。下面给出一个实时文语转换的例子:

输入文本: [n]jin1tian1[w]tian1qi4[w]hen3hao3[p],

其中 [n] 表示四级重音模式中的正常重音状态, [w] 表示一个词边界, [p] 表示一句话完后备暂停, 汉语拼音如 jin tian 对应于音节名, 汉语拼音后的阿拉伯数字表示音调 0, 1, 2, 3, 4 分别表示轻声、阴平、阳平、上声、去声。

上边的文本对应的汉语是“今天天气很好”。

合成器发出的声音流畅自然。例如: 每个词之间略有停顿; 双字词第 1 个字读得较重较长, 第 2 个字读得较轻较短; 一个词中 2 个字都是上声, 第 1 个字声调变为阳平等。

4 总结

现有的系统只是一个阶段性成果, 只完成了声学处理部分, 留有和语言学处理部分的接口, 完整的系统还应有语言学处理部分。所以进一步完善的措施包括针对汉语发音的特点, 选择一个自动分词系统, 该系统既要有较高的切分精度, 又要满足实时处理的要求, 即算法较简单。

系统中使用的规则库还有待于进一步充实, 规则算法有待于优化, 由于 C25 合成一个样点的平均时间只有 700 多条指令, 占一个取样周期的 7/10, 所以还可以将一部分规则运算移入 C25 以减轻主机负担。

现有系统是基于共振峰的参数编码方式, 由于汉语语言学对共振峰的研究还不够, 一些语言现象还没有在共振峰参数上得到一致的规则, 随着汉语语言学、语音学不断发展, 人们对共振峰的认识也必将逐渐加深, 它在规则合成方面的优势也将不断地得到挖掘。

汉语文语转换系统的完善还是一个漫长的过程, 它需要多方面的广泛的合作, 随着语音学、语言学、电子学、计算机科学的不断发展, 人们最终可以使这一系统得到完美的实现, 为人机语声对话提供基础。

5 参考文献

- 1 Klatt D H. Software for a cascade / parallel formant synthesizer. JASA, 1980, 67(3): 971
- 2 Klatt D H. Review of text-to-speech conversion for English. JASA, 1987, 82(3): 737
- 3 吕士楠, 周同春, 谢咏圭. 用 Klatt 合成器合成汉语的初步研究. 见: 黄泰翼 主编. 第 2 届全国人机语言通讯

- 学术会论文集. 北京: 科学出版社, 1992. 281 ~ 286
- 4 赵小杰, 谢咏圭. Klatt 共振峰合成器的实时化研究. 北京师范大学学报 (自然科学版), 1994, 30(2): 212
- 5 中国科学院声学研究所. TMS 320C 25 用户指南. 北京: 科高公司, 1991
- 6 中国科学院声学研究所. TMS 320C 25 - D 开发板用户指南. 北京: 科高公司, 1991
- 7 拉宾纳 L R, 谢弗 R W. 语音信号数字处理. 朱雪龙 等译. 北京: 科学出版社, 1983
- 8 吴宗济, 林茂灿. 实验语音学概要. 北京: 高等教育出版社, 1989

FIXED-POINT REAL-TIME KLATT SYNTHESIZER AND THE APPLICATION IN CHINESE TEXT-TO-SPEECH SYSTEM

Xu Ke^{1,2)} Xie Yonggui¹⁾

(1) Department of Radio Electronics, Beijing Normal University, 100875, Beijing, PRC; 2) Center
for Space Science and Application Research, Academic Sinica, 100080, Beijing, PRC)

Abstract Based on the research of the dynamic range of the cascade/parallel channels and a large number of experiments, single-double precision integer are used together to simulate the Klatt synthesizer's floating-point operation. The Klatt synthesizer has been improved in six aspects. The new synthesizer attains approximately the original precision and dynamic range, and provides a powerful kind of means for Chinese synthesis by rule, and makes it possible to realize real-time synthesis on TMS 320C25. A Chinese text-to-speech system has been developed on a 386 microcomputer with a high speed digital signal processing board TMS 320C25 - D. The intelligibility and naturalness of the system are increased by a consideration of five phonological rules.

Keywords formant; real-time; speech synthesis; text-to-speech