一种抽样率转换的简化算法

欧光艳[1], 刘国勤[2], 朱安珏[1]
(1. 中国科学院声学研究所东海研究站, 上海 200032; 2. 中国人民解放军某部, 湛江 524022)

摘要: 为了改变信号的采样率,需要对信号进行插值和抽取。但是,当插值倍数很大时,运算量非常庞大,这在硬件上将难以实现。提出了一种简化算法:分析输出与输入、滤波器系数之间的关系,再结合采样率转换的比值,对所需要的点进行乘累加运算,实验表明,达到了采样率转换的目的,大大降低了运算量。

关键词: 抽样率转换;时变滤波器;插值;抽取
中图分类号: TB566　　文献标识码: A　　文章编号: 1000-3630(2008)-02-0279-04

A simplified algorithm for changing sampling rate

OU Guang-yan[1], LIU Guo-qin[2], ZHU An-jue[1]
(1. Shanghai Acoustic Laboratory, Institute of Acoustics, Chinese Academy of Sciences, Shanghai 200032, China)

Abstract: Extraction and interpolation of signal is necessary to be performed for changing the sampling rate of signal. When the rate of interpolation is high, the calculation becomes burdensome and hard to be implemented with hardware. An algorithm for reducing the computation is proposed. Based on analyzing the relationship between the input signal, output signal and filter coefficients and combining with the rate of sampling rate changing the algorithm performs multiply-accumulate operation to the points needed. Experiments show that sampling rate changing can be implemented with great reduction of computation load by this method.

Key words: change of sampling rate; poly-phase filter; interpolation; extraction

1 引　言

在声信号源的信号信息处理中,需要引入多普勒频移,对抽样率进行转换即可达此目的。如果采用简单的先插值后抽取[1]的方法,运算量将非常庞大,这在硬件上难以实现[2]。本文提出一种简化算法,通过分析输出与输入、滤波器系数之间的关系,再结合抽样率转换的比值,对结果所需要留下来的点进行乘累加运算, 以达到抽样率转换的目的,从而大大降低了运算量。

2 抽样率转换算法

2.1 信号抽取

信号抽取通过降低抽样率以去掉多余数据。将 $x(n)$ 中每 M 个点中抽取一个,依次组成一个新的序列 $x'(n)$,即

$$x'(n)=x(Mn), -\infty<n<+\infty \tag{1}$$

通过对 $x'(n)$ 进行 DTFT,得到

$$X'(e^{j\omega})=1/M\sum_{k=0}^{M-1}X(e^{j(\omega-2\pi k)/M})$$

式中 $X'(e^{j\omega})$ 和 $X(e^{j\omega})$ 分别是 $x'(n)$ 和 $x(n)$ 的 DTFT。这样 $X'(e^{j\omega})$ 是原信号频谱 $X(e^{j\omega})$ 先作 M 倍的扩展再在 ω 轴上每隔 $2\pi/M$ 的移位叠加。因此,为了防止信号混叠,还需要在抽取之前先对 $x(n)$ 做低通滤波,压缩其频谱,然后再抽取。抽取的结构图如图 1 所示。

2.2 信号插值

通过将 $x(n)$ 中每相邻两个点之间补 L-1 个零,从而将抽样频率 f_s 增加到 L 倍,得到 $v(n)$,即

$$v(n)=\begin{cases}x(n/L) & n=0, \pm L, \pm 2L\cdots \\ 0 & \text{其他}\end{cases}$$

记 $x(n)$ 和 $v(n)$ 的 DTFT 分别是 $X(e^{j\omega_x})$ 和 $X(e^{j\omega_v})$,

收稿日期: 2007-09-15; 修回日期: 2007-12-28
作者简介: 欧光艳(1982-),男,硕士研究生,研究方向: 信号信息处理
通讯作者: 欧光艳, E-mail:worldsmile1982@126.com

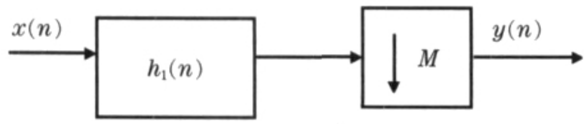


图 1 信号抽取的结构图
Fig.1 Schematic diagram for signal extraction

由于 $\omega_v=2\pi f/f_v=2\pi f/Lf_x=\omega_x/L$
可以得到

$$V(e^{j\omega_v})=X(e^{jL\omega_v})=X(e^{j\omega_x}) \tag{3}$$

上式说明，$V(e^{j\omega_v})$ 在 $(-\pi/L\sim\pi/L)$ 内等于 $X(e^{j\omega})$，这相当于将 $X(e^{j\omega})$ 做了周期压缩。插值之后，在原 ω_x 的一个周期内，$V(e^{j\omega_v})$ 变成了 L 个周期。因此，需要使用低通滤波器截取 $V(e^{j\omega_v})$ 以滤除多余的。所以插值的结构图如图 2 所示。

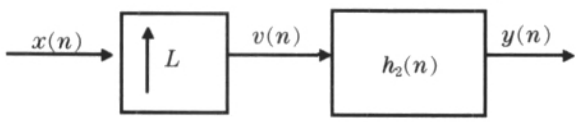


图 2 信号插值的结构图
Fig.2 Schematic diagram for signal interpolation

2.3 抽取与插值相结合的抽样率转换

为了将抽样率转变到 L/M 倍，为了不因为数据点的减少而造成信息的丢失，先对信号做插值，然后再抽取，如图 3 所示。

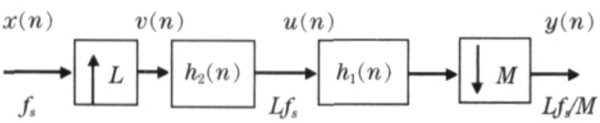


图 3 插值和抽取的级联结构
Fig.3 The cascade structure for signal extraction and interpolation

由图 3 可以看出，对输入 x(n) 先用插值器进行插值，然后经过滤波器，再通过抽取器，这样得到新的序列。由于插值后的信号的每一个点都要与滤波器的系数相乘，但结果是每 M 个点只要一个，因此有较多的乘法浪费，而当插值倍数很高（比如1000）时，计算量就变得十分庞大。现在我们来考虑它的简化问题。

图 3 中，$h_1(n)$ 和 $h_2(n)$ 所处理的信号的抽样率都是 Lf_s，因此可以将它们合起来变成一个滤波器 h(n)，如图 4 所示。由于

$$u(n)=v(n)*h(n)=\sum_{k=-\infty}^{\infty}v(n-k)h(k)$$

$$=\sum_{k=-\infty}^{\infty}h(n-Lk)x(k) \tag{4}$$

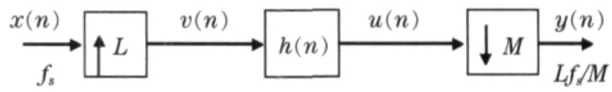


图 4 使用一个滤波器的信号插值与抽取的级联结构
Fig.4 The cascade structure for signal extraction and interpolation with a single filter

可以得出 y(n) 和 x(n) 的关系，即

$$y(n)=u(Mn)=\sum_{k=-\infty}^{\infty}h(Mn-Lk)x(k)$$

令 $k=[\frac{nM}{L}]-i$

则

$$y(n)=\sum_{i=-\infty}^{\infty}h(Mn-[\frac{nM}{L}]L+iL)x([\frac{nM}{L}]-i) \tag{5}$$

由于

$$Mn-[\frac{nM}{L}]L=Mn \mod L=\langle Mn\rangle_L$$

可以得到 y(n) 和 x(n) 的关系的表达式，即有

$$y(n)=\sum_{i=-\infty}^{\infty}h(iL+\langle Mn\rangle_L)x([\frac{nM}{L}]-i) \tag{6}$$

可以看出，对于输出 y(n) 的每一个点，我们可以通过将 x(n) 与 h(n) 中的对应系数进行乘累加，即可以得到该时刻的 y(n)。为了找到 h(n) 与 x(n) 的对应关系，可以将 y(n) 看作是 x(n) 通过一个时变滤波器[3]所得到的输出，记该时变系统的单位抽样响应为 g(n, m)，即

$$g(n,m)=h(nL+\langle Mm\rangle_L)$$
$$-\infty<n,m<+\infty \tag{7}$$

因为

$$g(n,m+kL)=h(nL+\langle Mm+kML\rangle_L)$$
$$=h(nL+\langle Mm\rangle_L) \tag{8}$$

所以 g(n, m) 是以变量 m 为周期的，周期为 L。

这样可以把的 L/M 倍的抽样率转换关系的输入、输出分别改写如下：

$$y(n)=\sum_{i=0}^{K-1}h(Mn+iL-[\frac{nM}{L}]L)x([\frac{nM}{L}]-i)$$

$$=\sum_{i=0}^{K-1}h(iL+\langle nM\rangle_L)x([\frac{nM}{L}]-i)$$

$$=\sum_{i=0}^{K-1}g(i+\langle n\rangle_L)x([\frac{nM}{L}]-i) \tag{10}$$

这样可以采用以下的抽样率转换结构，如图 5 所示。

可以发现，$f_{s2}/f_{s1}=L/M$，那么令 $t=f_{s2}/f_{s1}=L/M$，将图 6 中的 L/M 替换为 t，则可以直接实现 t 倍抽样率的

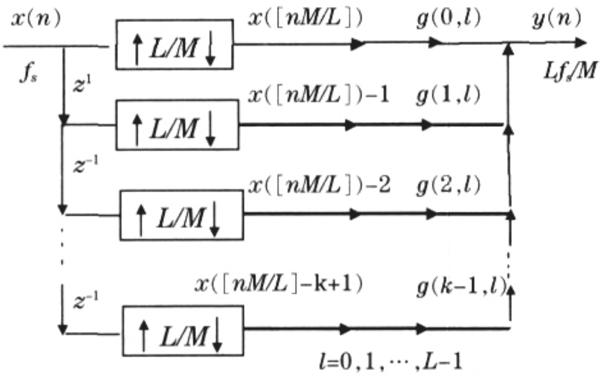


图 5　抽样率转换结构
Fig 5　Configuration for changing sampling rate

转换。

2.4　算法的 matlab 实现

当在计算机上实现 t 倍的抽样率转换时，可以采用图 6 的结构。

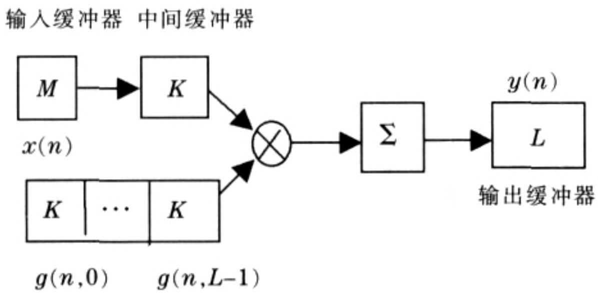
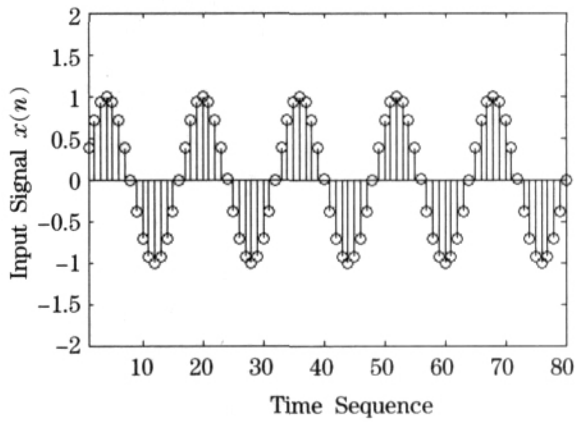
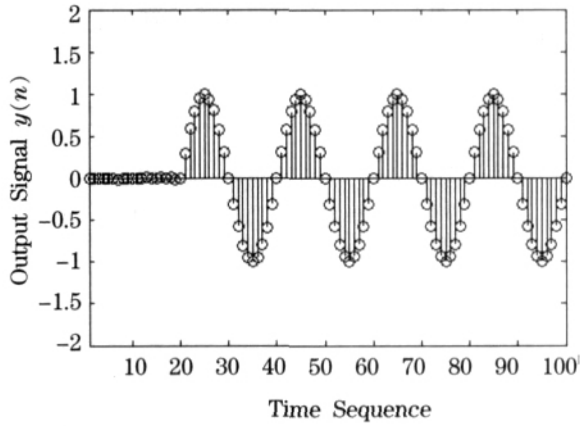


图 6　抽样转换的流程图
Fig 6　The flow chart for changing sampling rate



(a)输入信号



(b)输出信号 y(n)

图 7　输入信号 x(n)和输出信号 y(n)
Fig 7　Input signal x(n) and output signal y(n)

图中输入缓冲器的长度为 M, 中间缓冲器数据的长度为 K, 滤波器系数缓冲器共有 L 个, 每个系数缓冲器长度也为 K，输出缓冲器长度为 L，其中 t=L/M。

对于每一个输出数据块 y(n), n=0, 1……, L-1, 在每一个时刻 n, 一个子滤波器的 K 个系数和中间缓冲器的 K 个数据对应相乘, 然后相加, 得到该时刻的 y(n), 然后用下一个滤波器的系数和中间缓冲器数据相乘, 得到下一点输出 y(n)。当 nt 为整数时，从输入缓冲器移一个数据到中间缓冲器，当 L 个输出结束时, 应移进 Lt=M 个数据。如此重复, 则可得到整个抽样率转换的过程。

采用 matlab 仿真。假设 $x(n)=\sin(2\pi nf/f_s)$, 其中, 信号频率 f=1 000Hz, 抽样频率 f_{s1}=16 000Hz, t=1.25, 通过 matlab 仿真得到如图 7 的输入信号 x(n) 和输出信号 y(n)。

从图 7 可以看出, 与输入信号 x(n) 相比, 输出信号 y(n) 抽样率已经发生变化。x(n) 每周期抽样 16 次, 共 5 个周期, 而经过 t 倍抽样率变换后, y(n) 每周期抽样 20 次, 共 4 个周期, 则变换后的抽样率为信号频率 f 的 20 倍，抽样率变为 f_{s2}=f×20=1 000*20=20 000Hz。那么抽样率变换为 f_{s2}/f_{s1}=20 000/16 000=1.25, 与 t 相等, 符合预先的设计。

现在来分析下运输量减少的情况。假设输入序列长度为 n, FIR 滤波器的抽头为 N, 滤波器的运算复杂度记为 X(X=N 个乘法+(N-1) 个加法)。为了将抽样率提高到 t=L/M 倍, 如果采用先插值再抽取的方法, 则需要先插入 L 倍的信号, 再通过滤波器, 然后再以 1/M 的抽取率进行抽取, 那么运算的复杂度为 A=L*n*X。而采用本算法, 由于对滤波器进行了改造, 计算所需要的抽头个数减少为 N1=N/L, 输出信号长度为 n*L/M, 则其运算复杂度为 B=n*L/M*N1 个乘法+n*L/M*(N1-1) 个加法。因为加法的复杂度远小于乘法, 当输入序列很长(即 n>>N) 时,

$$B/A\approx\frac{n*L/M*N1}{n*L*X}$$

$$\approx\frac{n*(L/M)*(N/L)}{n*L*N}\approx 1/L/M$$

可以看出，当 L、M 都很大的时候，运算量将大大减少。例如，如果需要将抽样率由 1 000Hz 移到 1 001Hz，则令 L=1 000，M=999，那么运输量减少为 1/1 000*999≈10^{-6}。

3 结 论

本文提供了一种进行抽样率转换的简化算法。为了先插值再抽取的抽样率转换方法运算量太大的问题，本算法分析输出与输入、滤波器系数之间的关系，再结合采样率转换的比值，对所需要的点进行乘累加运算，以达到采样率转换的目的，大大降低了运算量，大大降低了系统资源的消耗。在声信号源信号信息处理系统中，利用 FPGA 制成的电路部分采用了该算法，从而成功地引入了多普勒频移。

参 考 文 献

[1] 胡广书. 数字信号处理-理论、算法与实现[M]. 第二版，清华大学出版社，2003.
HU Guangshu. Digital Signal Processing-theory, Arithmetic and Implement, Second edition[M]. Tsinghua University Press, 2003.

[2] Uwe Meyer-Baese. 数字信号处理的 FPGA 实现. 第二版[M]. 清华大学出版社，2006.
Uwe Meyer-Baese. Digital signal processing with field programmable gate arrays, Second Edition[M]. Tsinghua University Press. 2006.

[3] 李菊. 分数倍抽样率转换器的时变网络结构的 FPGA 实现 J]. 数据采集与处理，2005，20(3)：268-271.